DOCUMENT RESUME

ED 472 923 FL 027 548

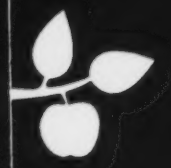
AUTHOR	Kjaersgaard, Poul Soren, Ed.
TITLE	Datalingvistik, 2000.
INSTITUTION	Odense Univ. (Denmark). Inst. of Language and Communication.
ISSN	ISSN-0906-7612
PUB DATE	2002-09-00
NOTE	100p.; Papers presented at the Annual Meeting of Datalingvistik Forening (10th, Odense, Denmark, May 22-23, 2000). Two papers are written in Danish.
PUB TYPE	Collected Works - General (020) -- Collected Works - Serials (022) -- Speeches/Meeting Papers (150)
JOURNAL CIT	Odense Working Papers in Language and Communication; n24 Sep 2002
EDRS PRICE	EDRS Price MF01/PC05 Plus Postage.
DESCRIPTORS	Age; *Computational Linguistics; *Danish; Databases; Foreign Countries; Morphology (Languages); Phonemes; *Phonetics; Pronunciation; Regional Dialects; Semantics; Uncommonly Taught Languages; Verbs
IDENTIFIERS	Denmark

ABSTRACT

Papers from the conference in this volume include the following: "Towards Corpus Annotation Standards--The MATE Workbench" (Laila Dybkjaer and Niels Ole Bernsen); "Danish Text-to-Speech Synthesis Based on Stored Acoustic Segments" (Charles Hoequist); "Toward a Method for the Automated Design of Semantic Representations" (Gregers Koch); "OLAM--et semiautomatisk morfologisk og lydstrukturelt kodningssystem for dansk" ("OLAM--A Semi-Automatic Morphological and Sound Structural Coding System for Danish") (Thomas O. Madsen, Hans Basboll, and Claus Lambertsen); "Semantic Encoding of Danish Verbs in SIMPLE--Adapting a Verb-Framed Model to a Satellite-Framed Language" (Bolette Sandford Pedersen and Sanno Nimb); and "En analyse af realisationer af fonemet [A:] i udvalgte regioner og aldersgrupper i Speechdat2" ("An Analysis of Realizations of the Phoneme [A:] in Selected Regions and Age Groups in Speechdat2") (Jacob Teckmeier and Kim Hornbech). (VWL)

Reproductions supplied by EDRS are the best that can be made
from the original document.

ED 472 923



UNIVERSITY OF SOUTHERN DENMARK

Datalingvistik 2000

Poul Søren Kjærsgaard (ed.)

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☑ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

PERMISSION TO REPRODUCE AND DISSEMINATE THIS MATERIAL HAS BEEN GRANTED BY

Steffen Nordahl

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

1

No. 24
September
2002

ODENSE WORKING PAPERS IN LANGUAGE AND COMMUNICATION

FL 027548

2

BEST COPY AVAILABLE

Datalingvistik 2000

Poul Søren Kjærsgaard (ed.)

Odense Working Papers in Language and Communication
No 24, September 2002

General editor: Steffen Nordahl Lund (snl@language.sdu.dk)

Institute of Language and Communication
University of Southern Denmark,
Campusvej 55
DK-5230 Odense M.
(lfn@language.sdu.dk)

Printed in Denmark by Syddansk Universitets Trykkeri

ISSN 0906-7612

Forord

Datalingvistisk Forenings 10. årsmøde fandt sted på Syddansk Universitet den 22. og 23. maj 2000. Institut for sprog og kommunikation samt undertegnede stod som arrangør af årsmødet.

I forlængelse af årsmødet har arrangøren påtaget sig det praktiske arbejde med at indsamle og redigere de skriftlige versioner af foredragene. Resultatet af dette arbejde er det her foreliggende.

Jeg vil gerne takke Institut for sprog og kommunikation for at have givet tilsagn om udgivelse af foredragene i sin skriftserie Odense Working Papers in Language and Communication (OWPLC). Men først og fremmest vil jeg rette en tak til bidragyderne for deres tålmodighed med at besvare mine spørgsmål i forbindelse med redigeringen. Og i den forbindelse vil jeg gerne undskylde den lange tid, som er gået siden indleveringen af manuskripterne. Det skyldes forhold, som er denne udgivelse uvedkommende.

I forbindelse med udgivelsen vil bidragene være at finde i pdf-format på følgende adresse http://www.sdu.dk/hum/psk/dalf2000/2000.html.
Fra denne adresse kan de enkelte bidrag hentes.

Odense, september 2002

Poul Søren Kjærsgaard

Indholdsfortegnelse

Laila Dybkjær, Niels Ole Bernsen : Towards Corpus Annotation Standards – The MATE Workbench 1

Charles Hoequist: Danish Text-to-Speech Synthesis Based on stored acoustic segments..... 21

Gregers Koch: Toward a Method for the Automated Design of Semantic Representations..... 33

Thomas O. Madsen, Hans Basbøll, Claus Lambertsen: OLAM – et semiautomatisk morfologisk og lydstrukturelt kodningssystem for dansk..... 43

Bolette Sandford Pedersen, Sanni Nimb: Semantic Encoding of Danish Verbs in SIMPLE – Adapting a verb-framed model to a satellite-framed language..... 57

Jacob-Teckemeier, Kim Hornbech: En analyse af realisationer af fonemet [A:] i udvalgte regioner og aldersgrupper i SpeechDat2 81

Towards Corpus Annotation Standards
The MATE Workbench[1]

Laila Dybkjær, Niels Ole Bernsen
Natural Interactive Systems Laboratory
Science Park 10, 5230 Odense M, Denmark
E-post: laila@nis.sdu.dk, nob@nis.sdu.dk

Abstract
This paper describes the European MATE project and its work towards speech corpus annotation standards. Focus is on the MATE workbench which is thoroughly described and illustrated.

1. Introduction
The aim of the European Telematics project MATE (Multilevel Annotation Tools Engineering) has been to facilitate the use and reuse of spoken language resources, coding schemes and tools by addressing theoretical issues and implementing practical solutions. MATE was launched in 1998 in response to the increasing need - not least in the area of spoken language dialogue systems research and development - for tools and standardisation efforts in support of efficient markup of spoken dialogue corpora at multiple levels.

The main results of the project are the MATE markup framework which bridges between the theoretical and the practical activities of MATE and is proposed as a standard for the definition and representation of markup for spoken dialogue corpora (Dybkjær and Bernsen 2000b), and the MATE workbench which supports the use of the markup framework. Even if the project has formally come to an end, the project consortium has continued funding for improving the workbench and new versions continue to appear. The newest version of the software can always be downloaded from the MATE web site at

[1] This paper is also being published in the Proceedings of the COCOSDA meeting in Beijing, 21 October 2000.

http://mate.nis.sdu.dk. Both an executable version and the source code is available, the latter under the GNU open source LGP license.

A discussion forum has been set up at the MATE web site for asking questions and sharing experience on the workbench, and for adding new tools to the MATE workbench to enhance its functionality.

In the following, Section 2 briefly describes the theoretical approach of MATE, Section 3 provides a detailed walkthrough of the MATE workbench, and Section 4 mentions related work.

2. Theoretical Background

To provide a solid basis for the coding standard to be proposed by MATE, more than sixty existing coding schemes belonging to five different coding levels (i.e. prosody, (morpho-) syntax, co-reference, dialogue acts, and communication problems) and their cross-level interaction were reviewed (Klein et al. 1998). The collected information served as background for establishing the MATE markup framework (Dybkjær et al. 1998, Dybkjær and Bernsen 2000b). In MATE, a coding level is some level of abstraction at which to conceptualise, tag, analyse, and retrieve information inherent in language corpora.

The MATE markup framework is a conceptual model which basically prescribes— (i) how files are structured, for instance to enable multi-level annotation, (ii) how tag sets are represented in terms of elements and attributes, and (iii) how to provide essential information on markup, semantics, coding purpose etc.

The central concept of the MATE markup framework is the coding module. A coding module is an extended kind of coding scheme which prescribes what constitutes a coding, including the representation of markup and the relations to other codings. Coding modules incorporate (i), (ii) and (iii) above. Thus, the MATE coding module is a proposal for a standard description of coding schemes.

For each of the above-mentioned five annotation levels and the issues to do with cross-level annotation, one or several of the reviewed coding schemes were adopted as starting points for the definition, following the MATE markup framework, of best practice coding schemes for implementation in the MATE workbench (Mengel et al. 2000).

3. The MATE Workbench

The MATE workbench (Isard et al. 1998, Isard et al. 2000, Dybkjær and Bernsen 2000a) is a software tool set which supports the MATE markup framework, incorporates the MATE best practice coding schemes, and enables users to annotate corpora and extract information about annotated corpora via a user-friendly interface. The workbench is in continued development and, although the full functionality and desired user-friendliness has not yet been achieved, there is already considerable interest among colleagues from around the world in using the MATE workbench.

The workbench is implemented in Java. It has been tested on Unix (Solaris) and Windows (NT and 98) but should run on any platform for which Java 1.2 or newer is available. The workbench has a modular architecture which facilitates the addition of new modules and new tool functionality by its users. XML is used for internal file representation. Style sheets are used for specifying the visual presentation of data to users. Style sheets are written in the MATE Style sheet Language (MSL). The emerging standard in this area is XSLT. XSLT, however, was not fully defined when the workbench was being designed, and lacked various necessary functionalities at the time. It was therefore decided to implement MSL which uses the MATE query language but is otherwise similar to XSLT.

In the following, we describe and illustrate the MATE workbench functionalities.

3.1. Getting started

As mentioned, the MATE workbench can be downloaded from the MATE web site at http://mate.nis.sdu.dk. The download page describes how to start the workbench from a command tool. Starting the workbench will result in the following two windows being opened. The first window (Figure 1) allows the user to open new project windows (Figure 2), access available tools, such as the coding module editor (Figure 5), and access the online help function.

The second window (Figure 2) shows example projects which include best practice coding schemes and annotated examples for a number of different annotation levels.

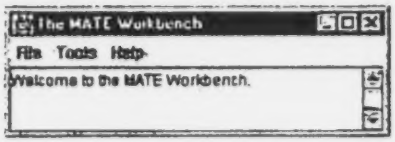


Figure 1. The control window.

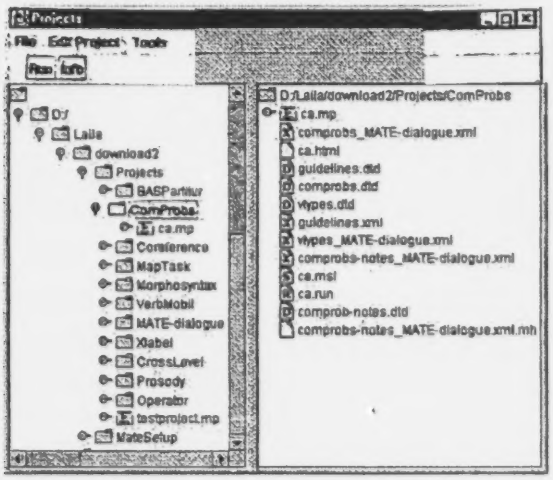


Figure 2. The projects window.

3.2. Annotation

The two main functionalities of the MATE workbench are support for annotation of spoken dialogue corpora and linguistic data, and information

BEST COPY AVAILABLE

extraction from annotated corpora. The workbench annotation support is described in the following.

3.2.1. Best practice coding schemes

Ready-to-use best practice coding schemes are included for most of the coding levels addressed by MATE. Prosody is not yet included. Style sheets allow the user to view and annotate files in an appropriate way using any of the implemented best practice schemes. Example dialogues are provided which illustrate how a tagged dialogue and the accompanying tag set will be shown when a particular best practice scheme is being applied. Internally in the workbench, an annotated dialogue is represented as a set of references to the transcription of the dialogue and possibly to other coding files. By default, the transcription refers to timeline information. The two windows in Figures 3 and 4 show annotation based on the MATE MapTask scheme and the MATE communication problems scheme, respectively. Figures 3 and 4 aptly illustrate the very different display requirements imposed by different coding schemes.

Clicking on the green 'PLAY' button in the Map Task window in Figure 3 will result in the audio file corresponding to the transcribed turn being played. The giver and the follower are the two speakers. Each speaker turn can be annotated with a speech act chosen from the list on the left by selecting the speaker and then clicking on the tag to be assigned. For example, the second giver utterance has been annotated with the 'instruct' dialogue act.

In the communication problems window in Figure 4, the orthographically transcribed dialogue is shown in the upper left-hand panel. To support the coder, guidelines for cooperative dialogue are shown in abbreviated form in the upper right-hand panel. Types of violation of particular guidelines are incrementally added by the coder in the lower right-hand panel. This panel is empty when annotation starts. The blue markup in the dialogue refers to the types of violation described in the lower right-hand panel and the violations themselves

refer to the guidelines. The lower left-hand panel shows annotator's notes. Again, this panel is empty when annotation starts. Notes can be added whenever the annotator needs to add an explanation of, e.g., why something went wrong in the dialogue so as to cause a communication problem.

There is no example coding module for transcription in the MATE workbench. Instead, a converter from Transcriber format (http://www.etca.fr/CTA/gip/Projets/Transcriber/) to MATE format enables transcriptions made using Transcriber to be exported to MATE format and annotated using the MATE workbench.

3.2.2. Adding a new coding module

Users may add new coding modules (coding schemes) themselves for existing or new coding levels via the coding module editor, cf. Figure 5. In order for a coding scheme and the dialogues annotated using it to be usable and understandable by people other than its creator, some key information must be provided. The MATE coding module which is the standard coding scheme description format proposed by MATE, serves to capture this information. A coding module includes the ten items shown in Figure 5. It prescribes what constitutes a coding, including markup representation and relations to other codings (module references).

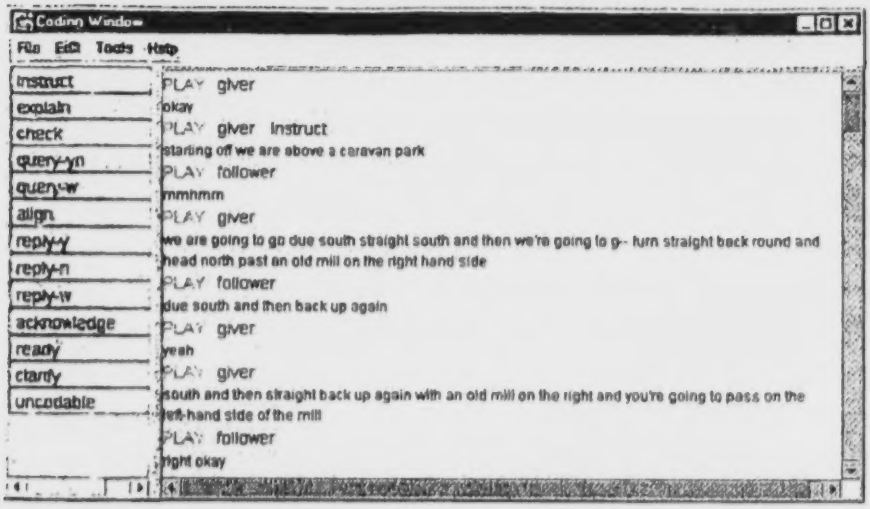


Figure 3. Annotation using the MATE Map Task scheme.

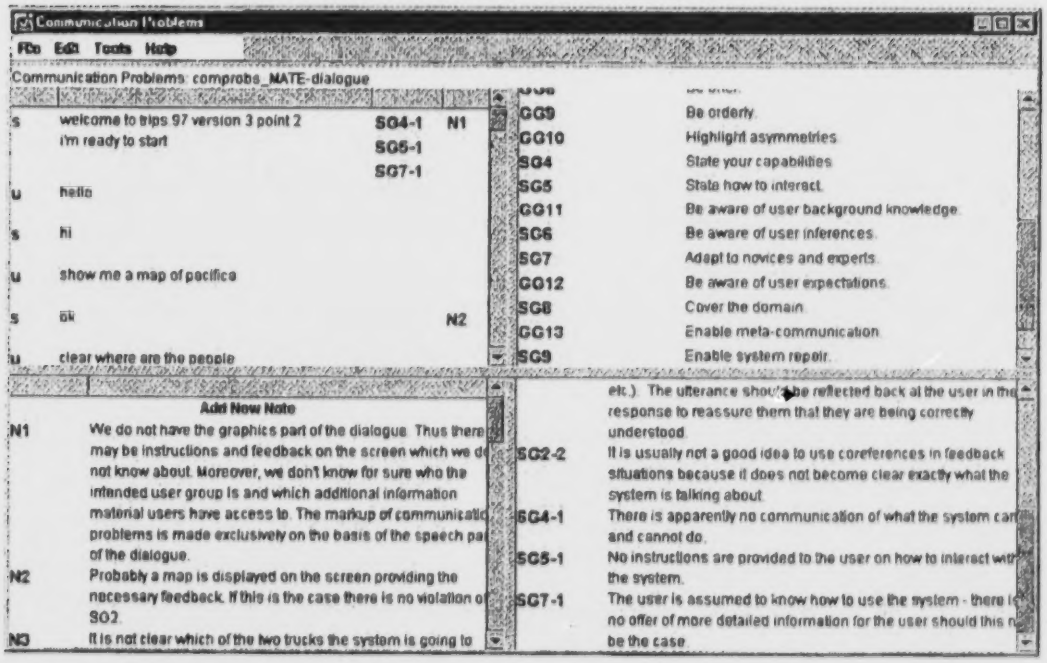

Communication Problems
File Edit Tools Help
Communication Problems: comprobs_NATE-dialogue
s welcome to trips 97 version 3 point 2
i'm ready to start
SG4-1 N1
SG5-1
SG7-1
u hello
s hi
u show me a map of pacifico
s ok
N2
u clear where are the people
GG9 Be orderly.
GG10 Highlight asymmetries.
SG4 State your capabilities.
SG5 State how to interact.
GG11 Be aware of user background knowledge.
SG6 Be aware of user inferences.
SG7 Adapt to novices and experts.
GG12 Be aware of user expectations.
SG8 Cover the domain.
GG13 Enable meta-communication.
SG9 Enable system repair.
Add New Note
N1 We do not have the graphics part of the dialogue. Thus there may be instructions and feedback on the screen which we do not know about. Moreover, we don't know for sure who the intended user group is and which additional information material users have access to. The markup of communication problems is made exclusively on the basis of the speech part of the dialogue.
N2 Probably a map is displayed on the screen providing the necessary feedback. If this is the case there is no violation of SG2.
N3 It is not clear which of the two trucks the system is going to
etc.). The utterance should be reflected back at the user in the response to reassure them that they are being correctly understood.
SG2-2 It is usually not a good idea to use coreferences in feedback situations because it does not become clear exactly what the system is talking about.
SG4-1 There is apparently no communication of what the system can and cannot do.
SG5-1 No instructions are provided to the user on how to interact with the system.
SG7-1 The user is assumed to know how to use the system - there is no offer of more detailed information for the user should this not be the case.

Figure 4. Annotation using the MATE communication problems scheme.

BEST COPY AVAILABLE

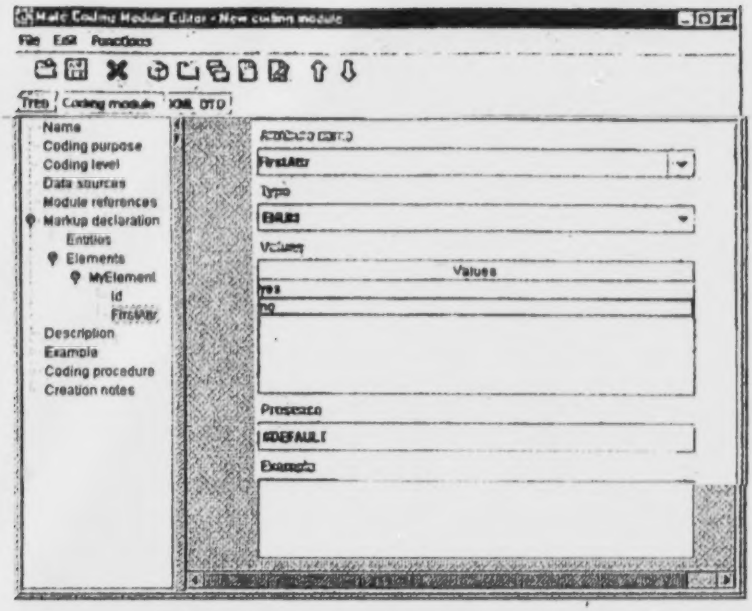


Figure 5. The coding module editor.

3.2.3. Adding a new project

A new MATE project folder can be created (Figure 6) via the 'File' menu in the project window in Figure 2.

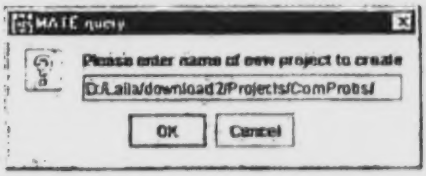


Figure 6. Creating a new project.

3.2.4. Adding a new file

Figure 7 shows the browsing for an existing file to be added to a project folder. The window is invoked from the 'Edit Project' menu in the project window in Figure 2. New files in a particular format may be created

BEST COPY AVAILABLE

(Figure 8) via the same menu and files may also be removed via this menu. As a minimum, a new XML file which is being created must be given a name, and the DTD or coding module to be applied must be specified. However, it is recommended that further header information is provided as well, cf. Figure 9.

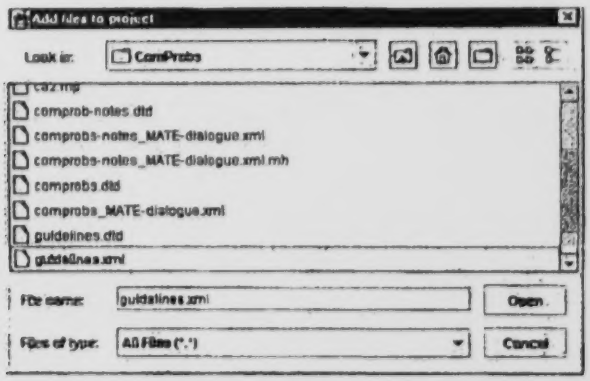


Figure 7. Browsing to add a file to a project folder.

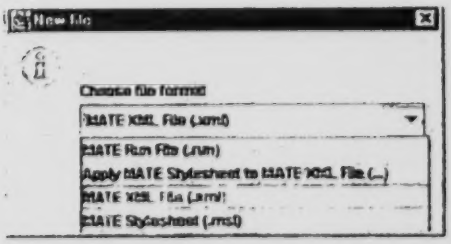

New file
Choose file format
MATE Run File (.run)
Apply MATE Stylesheet to MATE XML File (...)
MATE XML File (.xml)
MATE Stylesheet (.mst)

Figure 8. Creating a new file.

3.2.5. Listening to audio files

Audio files can be selected in the projects window (Figure 2) and played using the MATE audio tool. The window in Figure 10 will then appear displaying the sound file as a waveform. The audio tool might be used during transcription if a user has added an appropriate transcription coding

BEST COPY AVAILABLE

module. As it stands, the audio tool acts as a support tool during annotation and annotation review. For instance, when annotating communication problems there is sometimes a need for listening to the speech file in order to disambiguate an utterance and diagnose what went wrong.

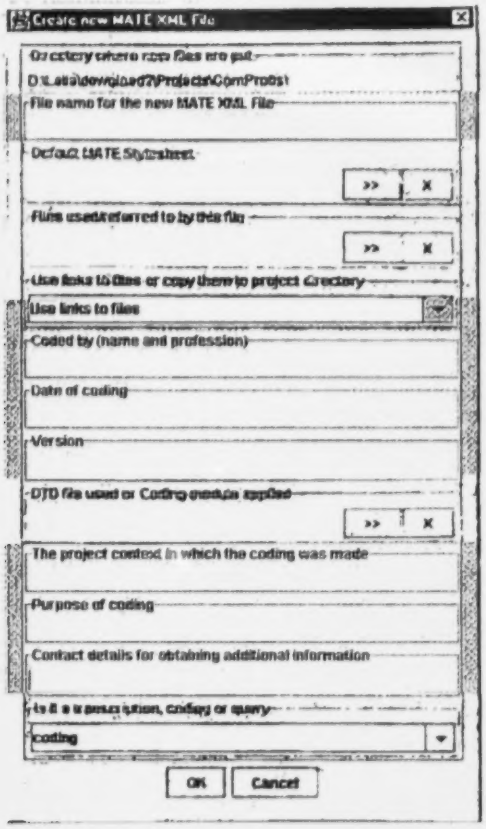

Create new MATE XML File
File name for the new MATE XML File
Default MATE Stylesheet
Files used/referred to by this file
Use links to files or copy them to project directory
Use links to files
Coded by (name and profession)
Date of coding
Version
DTD file used or Coding module applied
The project context in which the coding was made
Purpose of coding
Contact details for obtaining additional information
coding
OK
Cancel

Figure 9. Creating a new XML file

BEST COPY AVAILABLE

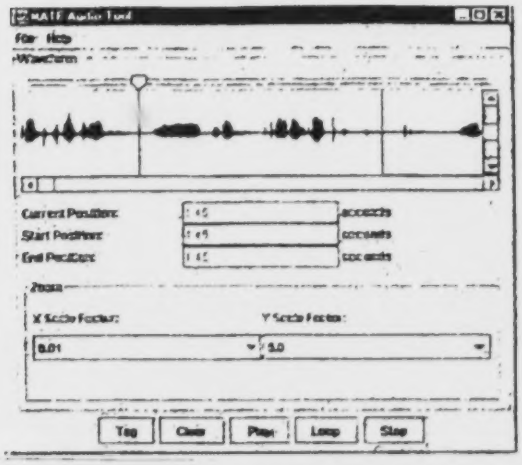


Figure 10. The MATE audio tool.

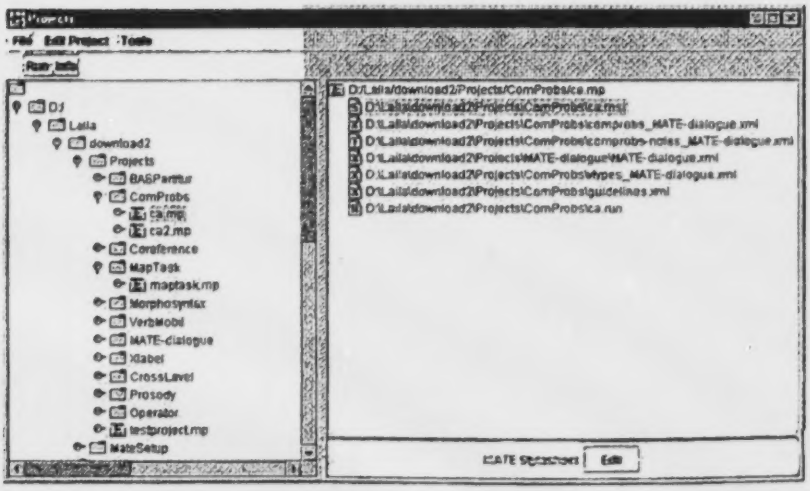


Figure 11. The edit option.

3.2.6. Editing a file

The MATE workbench uses three basic file types. Coding files are XML files. Style sheets are MSL files used for visualisation of codings and for annotation support. Runfiles specify which style sheet to apply to which XML file. Any XML, MSL or runfile in a MATE project folder can be

BEST COPY AVAILABLE

opened and edited (cf. Figures 12-14) by selecting the file and clicking on 'Edit' in the projects window, cf. Figure 11.

3.2.7. Import from and export to other file formats

Files may be imported from other formats to XML. For the moment, conversion from XLabels and BAS Partitur to XML is enabled. New converters can easily be added, including converters which export from XML to other formats, such as HTML. Figure 15 shows import from BAS Partitur.

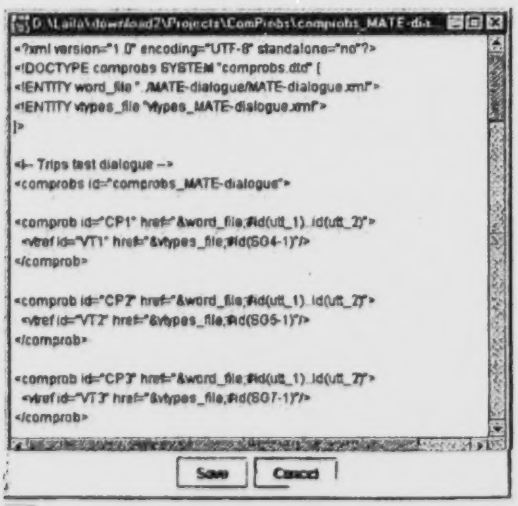


Figure 12. Editing an XML file.

BEST COPY AVAILABLE

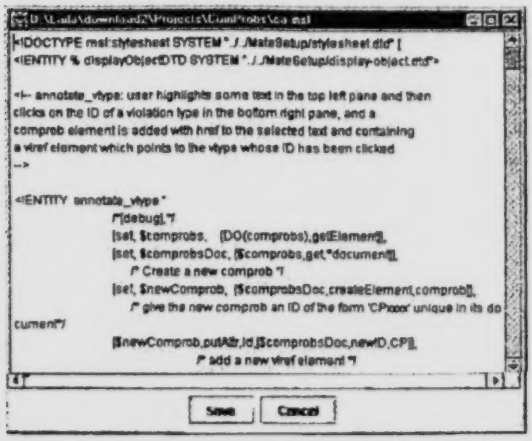

<!DOCTYPE mat:stylesheet SYSTEM "../../MateSetup/stylesheet.dtd" [
<!ENTITY % displayObjectDTD SYSTEM "../../MateSetup/display-object.mtd">
<!-- annotate_vtype: user highlights some text in the top left pane and then
clicks on the ID of a violation type in the bottom right pane, and a
comprob element is added with href to the selected text and containing
a vtref element which points to the vtype whose ID has been clicked
-->
<!ENTITY annotate_vtype "
/*[debug]*/
[set, $comprobs, (DO(comprobs),getElement)],
[set, $comprobsDoc, ($comprobs,get,"document")],
/* Create a new comprob */
[set, $newComprob, ($comprobsDoc,createElement,comprob)],
/* give the new comprob an ID of the form 'CProb#' unique in its do
cument*/
[$newComprob,putAttr,id,[$comprobsDoc,newID,CP]],
/* add a new vtref element */
Save
Cancel

Figure 13. Editing a style sheet.

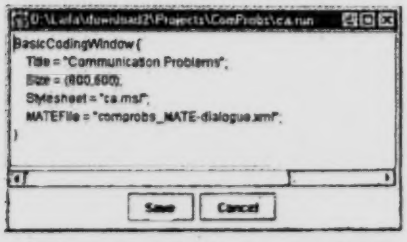


Figure 14. Editing a runfile.

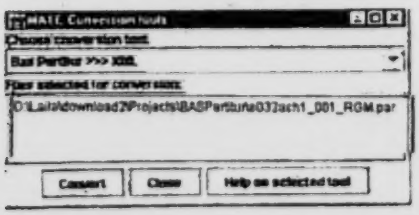


Figure 15. Import from BAS Partitur to XML.

3.3. Extracting information from annotated corpora

Once a corpus has been annotated, it must be possible to extract information from it for many different purposes. The MATE query tool is available for selecting the document(s) to be queried and for specifying the information to be subjected to a query, cf. Figure 16 (Isard et al. 2000). The query tool is activated from the tools menu in a coding window, cf. Figures

BEST COPY AVAILABLE

3 and 4. The information which can be extracted includes statistical information. Results are shown as illustrated in Figure 17. The interface for displaying results is not yet finalised. This is why the query results window for the moment only shows the raw XML data extracted from the queried XML file.

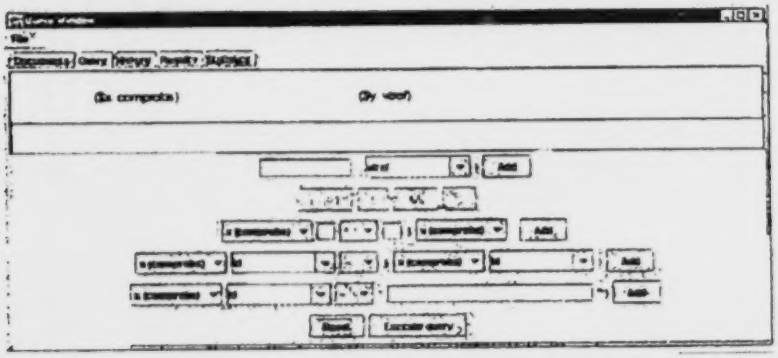


Figure 16. The query window.

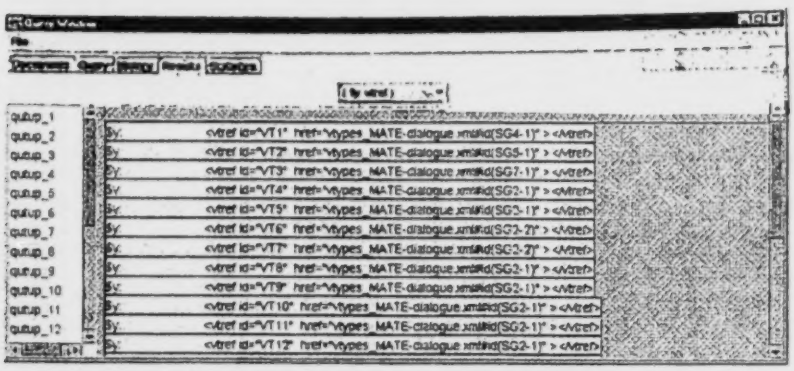


Figure 17. Results of a violation types query, cf. Figure 4.

BEST COPY AVAILABLE

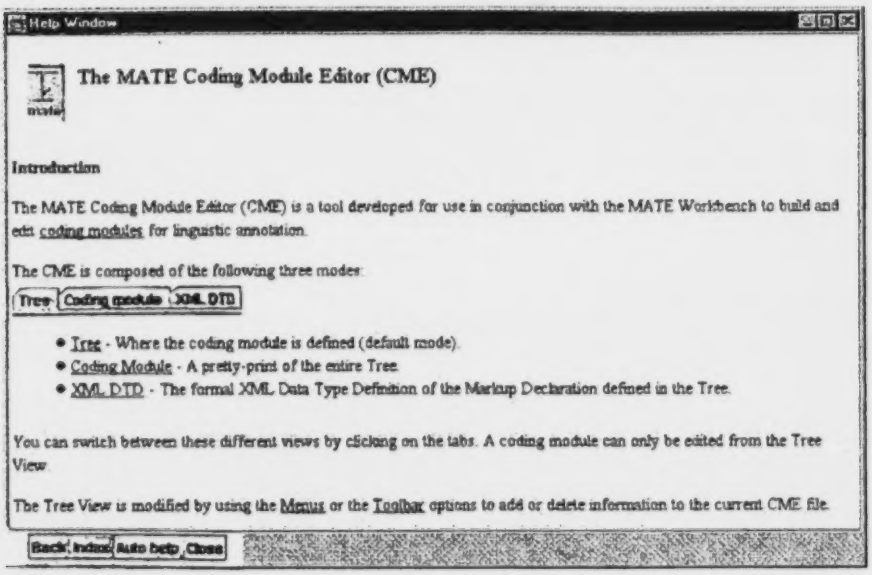

Help Window
The MATE Coding Module Editor (CME)
Introduction
The MATE Coding Module Editor (CME) is a tool developed for use in conjunction with the MATE Workbench to build and edit coding modules for linguistic annotation.
The CME is composed of the following three modes:
Tree
Coding module
XML DTD
Tree - Where the coding module is defined (default mode).
Coding Module - A pretty-print of the entire Tree.
XML DTD - The formal XML Data Type Definition of the Markup Declaration defined in the Tree.
You can switch between these different views by clicking on the tabs. A coding module can only be edited from the Tree View.
The Tree View is modified by using the Menus or the Toolbar options to add or delete information to the current CME file.
Back
Index
Auto help
Close

Figure 18. The help window.

3.4. Getting help

An online help facility may be consulted at any time during use of the MATE workbench. Figure 18 shows the topmost help page for the coding module editor. Explanation of particular coding schemes is available from the help menu in the coding window, cf. Figures 3 and 4.

3.5. Usability

Usability is a key concern in MATE. The focus on usability is reflected in the coding module concept and in the workbench coding module editor. The coding module prescribes a comprehensible standard description to be made of any coding scheme. The coding module editor makes it easy to specify coding modules – not least their markup declarations - also for non XML-literate users. The editor enables the user to specify the markup declaration for a new coding module almost without requiring any

BEST COPY AVAILABLE

knowledge of the underlying XML representation. The coding module editor automatically generates a DTD which is then used internally by the workbench. The coding module editor thus represents a major step forward compared to tools which require users to write DTDs.

There are still a couple of major usability issues to be solved, however. One issue is a user-friendly way of creating new coding visualisations. Writing style sheets for the workbench is cumbersome and requires programming skills because no editor is provided. The user must edit the raw style sheet code (or write new code), cf. Figure 13. It is high on our wishlist to enable users to easily define new visualisations. This may be done either by providing a style sheet editor comparable to the coding module editor as regards ease of use, or, alternatively, through a completely new interface concept replacing the need for style sheets and enabling users to easily define new visualisations.

A second major issue is the interface to the query tool and its results. As for the latter, Figure 17 makes it evident that usability improvements are needed. The query results could be presented far more transparently using an appropriate style sheet. This is also on the MATE to-do list. So is a more comprehensible interface for expressing queries than the present one (Figure 16).

4. State of the Art

Several frameworks for speech corpus annotation have been proposed but to our knowledge the MATE markup framework is still the more comprehensive framework around. An example of another framework is the annotation framework recently proposed by Bird and Liberman (1999)

which is based on annotation graphs. These are now being used in the ATLAS project (Bird et al. 2000) and in the Transcriber tool (Geoffrois et al. 2000). The annotation graphs serve as an intermediate representation layer between interface and internal data structures. Whilst Bird and Liberman do not consider coding modules or discuss the interface from a usability point of view, they present detailed considerations concerning timeline representation and timeline reference. The two frameworks may, indeed, turn out to complement each other nicely.

Acknowledgements

We gratefully acknowledge the support for the MATE project provided by the European Commission's Telematics/Language Engineering Programme. We would also like to thank all MATE partners. Without the very considerable joint efforts of the project consortium it would not have been possible to build the MATE workbench.

5. References

The MATE workbench is available in executable version and under the GNU open source LGP license from the MATE web site at http://mate.nis.sdu.dk. MATE reports are also available from this web site.

Bird, S. and Liberman, M. 1999. *A Formal Framework for Linguistic Annotation.* Technical Report MS-CIS-99-01. Department of Computer and Information Science, University of Pennsylvania.

Bird, S., Day, D., Garofolo, J., Henderson, J., Laprun, C. and Liberman, M.: ATLAS 2000. A Flexible and Extensible Architecture for Linguistic

Annotation. In *Proceedings of the 2nd International Conference on Language Resources and Evaluation* (LREC 2000). Athens:1699-1706.

Dybkjær, L. and Bernsen, N. O. 2000a. The MATE Workbench. *Proceedings of the LREC'2000 workshop on Data Architectures and Software Support for Large Corpora,* Athens: 33-37.

Dybkjær, L. and Bernsen, N. O. 2000b. The MATE Markup Framework. *Proceedings of the 1st SIGdial Workshop on Discourse and Dialogue,* Hong Kong.

Dybkjær, L., Bernsen, N. O., Dybkjær, H., McKelvie, D. and Mengel, A. 1998. *The MATE Markup Framework.* MATE Deliverable D1.2.

Geoffrois, E., Barras, C., Bird, S. and Wu, Z. 2000. Transcribing with Annotation Graphs. *Proceedings of the 2nd International Conference on Language Resources and Evaluation* (LREC 2000), Athens:1517-1521.

Isard, A., McKelvie, D., Cappelli, B., Dybkjær, L., Evert, S., Fitschen, A., Heid, U., Kipp, M., Klein, M., Mengel, A., Møller, M. B. and Reithinger, N. 1998. *Specification of Workbench Architecture.* MATE Deliverable D3.1.

Isard, A., McKelvie, D., Mengel, A., Møller, M. B., Grosse, M. and Olsen, M. V. 2000. *Data Structures and APIs for the MATE Workbench.* MATE Deliverable D3.2.

Klein, M., Bernsen, N. O., Davies, S., Dybkjær, L., Garrido, J., Kasch, H., Mengel, A., Pirrelli, V., Poesio, M., Quazza, S. and Soria, C. 1998. *Supported Coding Schemes.* MATE Deliverable D1.1.

Mengel, A., Dybkjær, L., Garrido, J., Heid, U., Klein, M., Pirrelli, V., Poesio, M., Quazza, S., Schiffrin, A. and Soria, C. 2000. *MATE Dialogue Annotation Guidelines.* MATE Deliverable D2.1.

Danish Text-to-Speech Synthesis Based on stored acoustic segments

Hoequist, Charles
Center for PersonKommunikation, Aalborg University,
Fredrik Bajers Vej 7, DK-9220 Aalborg Øst, Denmark
E-post: ch@cpk.auc.dk

Abstract

As part of a Danish Research Ministry strategy for speech technology R&D, a text-to-speech system has been developed for Danish. In this paper we will look at that part of the initiative which is concerned with synthetic speech. The system used relies on a database of encoded speech segments. Advantages and disadvantages in comparison to signal generation by calculation are discussed.

1. Background

A report [1], prepared under the auspices of the Ministry of Research shows that the synthesis systems used in Denmark today are of such low quality that even understanding the synthesized output requires user training. Obviously, this rules out many promising applications.

For instance, only a limited number of Danes with visual impairments are able to use such equipment for interpreting printed material. People whose sight loss occurs in middle age or later will in all probability not be able to learn to use currently available equipment. Dyslexic, aphasic or illiterate users suffer a similar disadvantage.

The development of high-quality Danish synthetic speech is, however, crucial to the Government's IT policy of making the information society available to all. The Research Ministry therefore entered into a contract with a consortium made up of the Center for PersonKommunikation (CPK), Aalborg University, Aalborg; The Institute for General and Applied Linguistics (IAAS), Copenhagen University, Copenhagen; Tele Danmark A/S (TDK), Taastrup; Tawido ApS (TAW), Aalborg, and Dansk

Taleteknologi A/S (DT), Aalborg. The two university partners - IAAS and CPK - are undertaking research within language and digital speech processing, respectively. It is the CPK work that is covered in this paper.

2. Synthesis System Overview

2.1. System architecture

The architecture of the Text-to-Speech system is as shown in Figure 1, see also [2].

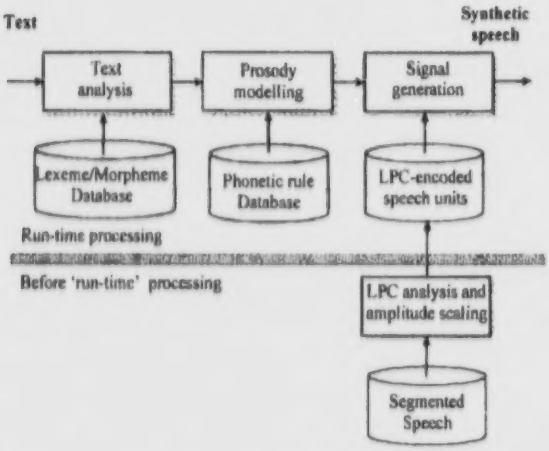


Figure 1. Architecture of the text-to-speech system

Input consists of two parallel streams: text and a subset of MS SAPI tags (see below) which are supported by DST (Dansk Syntetisk Tale).

The process of moving from a standard Danish orthographic input to acoustic output is handled in three stages: text preprocessing and analysis, prosody assignment and sound generation. Note that this is a process description; the corresponding software architecture is covered in section 4.

BEST COPY AVAILABLE

2.2. Text normalization and analysis

The first step in handling the text is preprocessing to turn all non-lexical forms (digits, abbreviations) into alphabetic representations of the intended spoken string. The output then goes to the text-analysis module, which performs both a morphological and a syntactic breakdown of the input. Words may be accessed as full lexical items (this is always the case for user-added items), broken down into component morphemes, or spelled out. Where the acoustic output is affected by syntactic factors, the output symbol string can include tags, e.g. where a syntactic boundary can trigger a pause.

2.3. Prosody assignment

The output string from the text analysis module is passed to the prosody assignment module, which assigns default duration and F0 values to each segment, as well as any pauses or F0 slope changes resulting from text-analysis tags or the presence of stød. Stød in this system is not a prerecorded diphone, but rather a sudden, sharp pitch drop, the acoustic realization of the stød's glottal constriction.

2.4. Sound generation

At this point in many systems, in particular older ones, the segment labels and prosodic instructions would be interpreted in terms of formant parameters, such as target frequencies, amplitudes and slopes, as in the Klatt [3] and Holmes [4] formant synthesizers. The parameters drive the generation of periodic and aperiodic signals to mimic the acoustics of human speech. This synthesis principle is still used in many commercial synthesizers today, not least because of the small memory footprint and the great freedom of control of the acoustic output. However, the quality of the output speech depends heavily on the rule set, which itself takes considerable time and expertise to develop. While copy-synthesized utterances using formant synthesizers show that extremely high quality is in principle possible, rule-based systems available to date are unable to approach this quality.

With the steady drop in cost for computer storage, concatenative synthesis as an alternative to rule-based generation is becoming more popular. Since concatenative synthesis relies on a pre-recorded speech database, the quality (particularly of the voice source) has the potential to be at least as high and generally much higher than rule-based generation. The DST system makes use of such a database.

The output of prosody assignment serves as instructions for selecting which RELP-encoded (Residual Excited Linear Prediction) diphones from the database are to be concatenated. The concatenated sounds are modified in accordance with any F0 or rate modifications passed along in the prosody module output.

In the course of system development, considerable effort has gone into optimizing the database. Some optimizations are purely computational, as in the development of faster search procedures for diphones. Others, which attempt to lower the number of diphones in the database, depend for their acceptance on users' judgment of the quality of the resulting output.

For example, the implementation of stød as a runtime signal adaptation allowed the elimination of some 1200 stød-containing segments from the diphone database, reducing it to its present size of 2600 diphones, with a corresponding reduction in footprint, with little or no loss of intelligibility or quality for listeners [5].

However, an attempt at further storage reduction by creating short vowels from their long counterparts resulted in an overall loss of intelligibility [5], despite the apparent similarity of the short and long vowels' formant structures.

3. Design and Implementation

3.1. Interfacing to applications

Given the desire for a commercially feasible system, the Ministry contract specifies that the synthesis system is to be compatible with Microsoft's Windows 95/98/NT operating systems and that it be usable with a wide range of existing and future applications, e.g. screen readers and internet browsers. It was therefore decided to implement the DST system to support Microsoft's Speech Application Programming Interface (MS SAPI). The system is called from applications as shown in Figure 2. This design makes it possible for existing third-party applications which already use

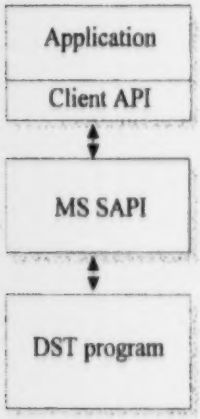


Figure 2. DST interface to MS SAPI compatible applications

MS SAPI for access to and control of a TTS for languages other than Danish to make immediate use of the DST system for Danish TTS. The DST system can also be used from C/C++, Visual Basic or OLE Automation, since MS SAPI complies with OLE COM (Object Linking and Embedding, Component Object Model).

DST system development plans include maintaining compliance with all aspects of future MS SAPI releases

which are relevant to DST functionality. This offers developers of TTS products a stable and publicly-available interface, which in turn increases the likelihood of significant commercial distribution for DST.

In order for the program to be usable with MS SAPI, it is implemented as a DLL (dynamic link library) for running under Windows. In addition, DST relies on an initialization file and databases with language-specific rules and diphones.

At the same time, the DST system has avoided building in Windows dependencies in the code wherever possible. The only platform-dependent module is the interface to MS SAPI (see following section). Use of DST under other operating systems is therefore feasible without too much effort.

3.2. Identification of modules

The synthesizer can be broken down to a number of individual modules as shown in Figure 3. This division makes it possible for parallel development of individual modules and their separate testing before integration into the product.

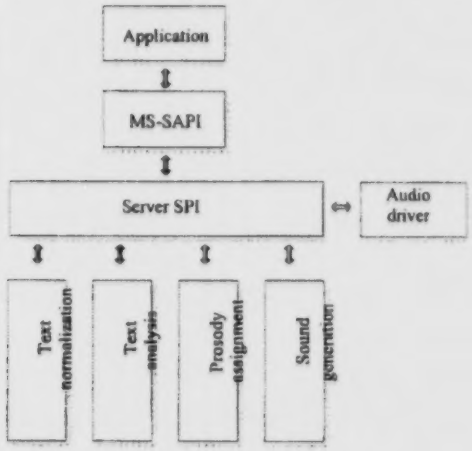


Figure 3. DST module architecture

Briefly, the modules are:

BEST COPY AVAILABLE

- **SPI** (Service Provider Interface), which serves as a connection between MS SAPI and the other modules in the system. The SPI interprets queries from applications (*Clients*), creates sentences out of a section of blocks of text and calls the underlying modules. MS SAPI tags are sent as a parallel stream via the processing interface, where every module in the processing path can inspect them for possible activity required in the particular module.
- **Text normalization,** which converts recognized abbreviations, dates, telephone numbers, etc. to their orthographic full forms.
- **Text analysis,** which maps orthographic strings to entries in the lexicon wherever possible.
- **Prosody assignment**, which calculates and annotates duration and pitch (F0) changes for the individual segments.
- **Sound generation,** which concatenates stored diphones to create an output audio stream.
- **Audio driver,** installed as a part of the Windows OS, is used to play out the synthesized speech signal.

4. Quality Measures

The foremost reason for the Research Ministry to initiate the project was to support the development of a high-quality Danish text-to-speech product. Quality is in this context to be interpreted as intelligibility and naturalness. The first commercial version of the synthesizer will at least be on a par with a demo version, which has been assessed as described below [6].

4.1. Intelligibility

The intelligibility test included synthetic speech from the DST system and as a reference natural speech.

The 32 test subjects listened to a total of 1600 words from the two categories. Each word was embedded in a carrier sentence: "*Der er [test word] de siger*" ("*It is [test*

word] they are saying") The percentages of words misheard are illustrated in Figure 4.

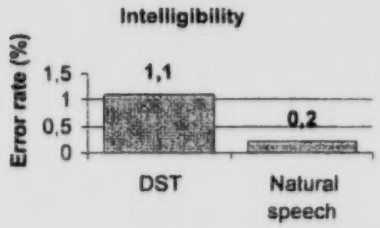


Figure 4 Word error rates for DST and natural speech

As expected, natural speech comes out with the highest intelligibility with an error rate of 0.2%, However, the DST system is also demonstrating high performance with only 1.1% errors.

4.2. Naturalness

The naturalness of the system was assessed by asking the same 32 subjects used in the above test to evaluate the naturalness of an utterance on a MOS (Mean Opinion Score) scale with values from 1 to 5. The higher the MOS is the more natural the utterance is. The test subjects were given speech from three different categories: natural speech, synthetic speech produced by the Infovox system 230, synthetic speech produced by the DST system. Figure 5 summarizes the results of the naturalness test. The naturalness of the DST system comes out with a score of 2.29, roughly midway between the Infovox score of 1.1 and a real speech score of 4.63.

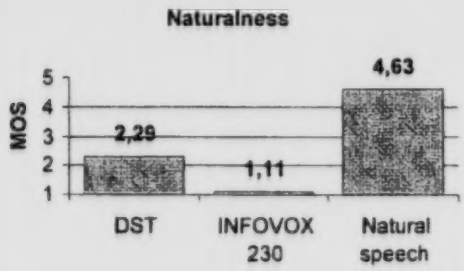


Figure 5: Naturalness (MOS) of DST, natural speech and Infovox

BEST COPY AVAILABLE

5. Future Plans

The project is moving ahead with plans for further quality improvements. These fall into two categories: first, the removal of artifacts based on the synthesis method, and second, a better and more robust modeling of natural speech as derived from text.

The primary artifact in concatenated synthesis is of course the potential for a discontinuity in the signal at every concatenation point. This is currently addressed by the preprocessing stage, where concatenation points are placed in low-amplitude sections of the signal. Work is underway to investigate the value of various types of signal smoothing at run time as well.

Modeling of natural speech occurs at various stages of the system. A later release will improve both text rules, to handle misspelled input, and the current prosody rules, to better model the pitch contour and pause structure of utterances. The possibility of expanding the diphone inventory to handle voicing variation in realizations of the Danish /r/ is being investigated, as well as some re-recording of the base speech for the diphone encoding. The re-recording is intended to address gaps in the original recordings, which did not adequately account for presence or absence of stress on a target diphone.

An additional area of research is the construction of the segment databases themselves. Creating a segment database requires less specialized knowledge and experience than the development of a rule system for formant synthesis. The disadvantage is that the database is dependent on the original recordings, and any segments not present there, or present but with low quality, usually require a new recording session and rebuilding of the database. This places a premium on having as much recorded and tagged material as possible to choose from. Unfortunately, the available speech databases are not geared toward this need. Most are designed to supply material for benchmarking speech recognizers, and there is little tagging of the kind common in text databases, where the database contexts are tagged. Speech databases would become very useful for concatenative synthesis if tagged with transcriptions and even analysis parameters.

6. Conclusions

The Danish Research Ministry has concluded that Danish synthetic speech is lagging behind comparable countries in quality and degree of market penetration.

To remedy this situation the Ministry has given a consortium the task to develop a new generation of Danish text-to-speech engines. Specific quality measures in terms of intelligibility and naturalness are intended to ensure that the system will surpass what is now available for Danish. The synthesizer is compliant with the MS SAPI interface. Hence, many applications as for instance screen readers, talking e-mails etc. can immediately be used together with the synthesizer. The above factors together with an attractive pricing will ensure another Ministry objective: widespread deployment of high-quality Danish text-to-speech.

7. References

[1] Dansk Syntetisk Tale 1996 (februar). Udarbejdet for Forskningsministeriet af Hjælpemiddelinstituttet.

[2] Jensen, J., Nielsen, C., Andersen, O., Hansen, E., and Dyhr, N.J. 1998.. A Speech Synthesizer with Modelling of the Danish "Stoed". In IEEE Nordic Signal Processing Symposium (NORSIG '98). Vigsø, Denmark.

[3] Klatt, D. H. 1980. Software for a cascade/parallel formant synthesizer. *Journal of the Acoustical Society of America*, 67:971-995.

[4] Holmes, J.N. 1985. A parallel-formant synthesizer for machine voice output. In F. Fallside and W.A. Woods (eds.). *Computer Speech Processing*. London:Prentice-Hall: 163-187.

[5] Andersen, O., Dyhr, N.-J., Nielsen, C. 1999. On Synthesizing Danish Short Vowels. In *Proceedings of the XIVth International Congress of Phonetic Sciences* (ICPhS '99). San Francisco:2291-2294.

[6] Bagger-Sørensen B. (1997). Testrapport (ter) for FRITSYN (Lyttetest) version 1.0. Tele Danmark, Udviklingsområdet.

Toward a Method for the Automated Design of Semantic Representations

Gregers Koch
Departement of Computer Science (DIKU)
University of Copenhagen
E-post: gregers@diku.dk

1. Automated Semantic Induction

Information is a concept of crucial importance in any conceivable scientific endeavour. Also the modeling of information or information modeling is becoming more and more important, not least in information systems. Databases, knowledge bases as well as knowledge management systems are continually growing. Modeling helps to understand, explain, predict, and reason on information manipulated in the systems, and to understand the role and functions of components of the systems. Modeling can be made with many different purposes in mind and at different levels. It can be made by emphasising the users' conceptual understanding. It can be made on a domain level on which the application domain is described, on an algorithmic level, or on a representational level. Here the interest is focused on modeling of information on a representational level.

As a logic programming activity, we are studying certain computational linguistic problems, in particular concerning the automation of data flow analysis synthesizing a sort of data flow structures. This automated method may be utilized for inductive purposes (in the sense of generalization from examples). The method here is called logico-semantic induction and it constitutes an efficient kind of automated program synthesis.

More precisely we are studying data flow structures and the construction and testing of software for automatic implementation of the principles of logico-semantic induction [Koch 1988, 1991, 1993]. The software seems to apply conveniently for the automated construction of 1) natural language interfaces to

knowledge based systems, 2) simple translation between human languages like English, Danish, and Japanese, 3) compilers of traditional programming languages, and 4) translators between fragments of human languages and logical formalisms.

1.1. An Introduction to the System

When constructing natural language interfaces, one necessarily has to select a linguistic fragment or a sublanguage to be used by the user in the communication or interaction with the computational system [Abramson and Dahl 1989]. Precisely which sublanguage or fragment should be selected, is an open question. Furthermore, which internal representation should be preferred from the point of view of efficiency or practical convenience, is also an open question [Pereira 1987]. Hence it may be a good idea to construct a frame system allowing flexible experimentation with language constructs and their possible representations. What is needed, is a facility to help in the automated translation of the selected constructs into some flexible and useful representations allowing further processing (e.g. for translation into a database query language or into an implemented programming language).

Here we discuss such a flexible home-made frame system. It is functioning in an inductive manner, since the system requires the user to specify a text or query combined with the suggestion for an internal representation. In this case the system is capable of working out the details of a translator system translating texts of a syntactic form similar to the given text into the internal representation in use. In other words, this is a method for inductive and automated program synthesis, sometimes called logico-semantic induction. Such a system may also be used for obtaining completely automatic implementations of parsers implementing semantic theories like Discourse Representaion Theory and Situation Semantics.

For a beginning let us give a closer description of the mentioned method for inductive and partly automated construction of programs for logical analysis of natural language texts [Koch 1992]. In an earlier paper the same method was applied to a scientific abstract [Koch 1997], and in a recent paper the same method was applied to Discourse Representation Theory [Koch 1999].

We shall illustrate the method by dealing with an utterly simple example. Here we shall analyse a tiny little sentence of four words

Peter seeks a mermaid

This sentence can trivially be described by means of the following syntax (a simple context-free grammar):

S → NP VP.
NP → Prop | D N.
VP → IV | TV NP.

The semantics in the form of a semantic parser, that is a program translating into some semantic representation, may be given in the form of a definite clause grammar (short DCG) like this

S(Z) → NP(X,Y,Z), VP(X,Y).
NP(X,Y,Y) → Prop(X).
NP(X,Z,W) → D(X,Y,Z,W), N(X,Y).
VP(.....) → TV(.....), NP(.....). (*)

with some lexical data

D(X,Y,Z,a(X,Y,Z)) → [a].

TV(X,Y,seeks(X,Y)) → [seeks].

N(X,mermaid(X)) → [mermaid].

Prop(Peter) → [Peter].

The last line (*) of the program was missing, it still needs to be filled out.

If we write it this way

VP(X,W) → TV(X,Y,Z), NP(Y,Z,W).

we obtain the extensional reading.

On the other hand, if we fill it out this way

VP(X,Z) → TV(X,Y,Z), NP(_,_,Y).

we obtain an intensional reading, where the existence of a mermaid is not presupposed.

A central point of this analysis is the observation, that the construction of this kind of parser programs can relatively easily be done automatically by another program, sometimes called the meta parser. The central part of such a meta parser may conveniently be a kind of data flow analysis, but other possibilities are also conceivable.

As an application, let us return to and continue with a closer analysis of the extensional reading of the small example sentence. In short, by interpreting a(X,Y,Z) in different ways, we obtain a number of different semantic representations.

For example, interpreting it this way

$a(X; Y; Z) = \exists X[Y \,\&Z]$

will give a predicate logic expression as the obtained semantic representation. On the other hand, interpreting it this way

$a(X; Y; Z) = [[X]; Y \,\&Z]$

will lead to a representation in the style of Discourse Representation Theory [Kamp and Reyle 1993].

2. Automated Design of Semantic Representation

We shall discuss a couple of new methods for extracting the informational content of (relatively brief) texts formulated in natural language (NL). It makes sense to consider information extraction from NL texts to be essentially the same task as building simple kinds of information models when parsing the texts. Here we present two new methods that are distinguished by extreme simplicity and robustness. The simplicity makes programming of the methods feasible, and so a kind of automatic program synthesis is obtained. The robustness causes wide applicability, and so the methods have a high degree of generality. We have to distinguish between two cases (or two situations).

Case 1: Let us here assume that we are in possession of a suggestion for the form of the information models. This means, we have a formalism for mapping into. In this case we can generate the translating parsers automatically from carefully selected examples, by means of so-called logico-semantic induction, as described for instance in.

Case 2: As a contrast, let us here assume that we do not know of any suggestion for the form of the information models. In this case we may follow the following recipe and hereby obtain a kind of semi-automatic design of consistent information structures.

In the simplest version of this analytic method, our text should be exposed to the following selectional or constructive steps.

Step 1: Select an example text.

Step 2: Construct a syntactic description, sufficient to describe the example text.

Step 3: Perform a complete data flow analysis on the example text to obtain an augmented syntactic tree structure (sometimes called a data flow structure).

Step 4: Extract a definite clause grammar (DCG) from the data flow structure.

Step 5: Supplement the DCG grammar obtained in step 4 with relevant lexical information.

Step 6: Perform a traditional kind of symbolic execution on the logic program obtained from step 4 and step 5.

Step 7: Extract all possible symbolic equations from the symbolic execution in step 6.

Step 8: Solve the symbolic equations to obtain a feasible solution. This solution constitutes a feasible representational structure.

Step 9: Perform logico-semantic induction, similar to case 1 above, to the text of step 1 combined with the solution of step 8.

Step 10: Perform a forward test run on the example from step 1, to compare the result with the solution of step 8.

Step 11: Perform a backward test run on the solution from step 8, to compare the result with the original text from step 1.

Step 12: Perform an accumulation of the acceptable fractional linguistic descriptions in the shape of definite clause grammars obtained through this process from a variety of carefully selected example texts.

In a tentative comparison with some important alternative approaches, it makes sense to distinguish between the design of representation, manual programming, and automated program synthesis.

3. Discussion

Hans Kamp and followers seem to handle the design of representation very well. I do not know if they can produce the hand-coded programs. Roger Schank and followers seem to be able to handle both the design and the hand-coded programming. But we should notice that their software solutions seem to suffer from mediocre reproducibility. As to other approaches, including that of professor Kawaguchi and others, they can handle the design of representation, but they seem unable to produce the hand-coded translation software. Some other alternatives were discussed in earlier papers. None of the mentioned alternative approaches seems to be able to produce the automated program synthesis. In con-

trast, my project reported here seems to handle the design of representation rather well and to give partial solutions to the manual programming (steps 3-8) and the automated program synthesis (steps 9-11 and case 1). In short, the method of case 1 and in particular the method of case 2 seem to be unique compared to today's scientific literature.

4. References

H. Abramson and V. Dahl 1989. *Logic Grammar*, Springer.

C. G. Brown and G. Koch, eds.) 1991., *Natural Language Understanding and Logic Programming*, III, Amsterdam:North-Holland.

Kamp, H. and Reyle U. 1993. *From Discourse to Logic*. Amsterdam:Kluwer.

Koch, G. 1993. Montague's PTQ as a Case of Advanced Text Comprehension, In H. Kangassalo et al. (eds.) *Information Modelling and Knowledge Bases IV*, Amsterdam:IOS:377-387.

G. Koch, 1992. Logics and informatics in an integrated approach to natural language database interfaces. In S. Ohsuga et al. (eds.) *Information Modelling and Knowledge Bases III*. Amsterdam:IOS:602-616.

G. Koch, 1991. Linguistic data flow structures. In Brown and Koch, op.cit.:293-308.

G. Koch, 1988. Computational logico-semantic induction. In V. Dahl and P. Saint-Dizier (eds.). *Natural Language Understanding and Logic Programming II*. Amsterdam:North-Holland:107-134.

G. Koch, 1997. Semantic analysis of a scientific abstract using a rigoristic approach. In H. Kangassalo et al. (eds.). *Information Modelling and Knowledge Basis VIII*. Amsterdam:IOS:361-370.

G. Koch, 1999. Discourse representation theory and induction. In H.C Bunt & E.G.C. Thijsse (eds.). *Proceedings of the Third International Workshop on Computational Semantics* (IWCS-3), Tilburg University:401-403.

F.C.N. Pereira and S.M. Shieber, 1987. *Prolog and Natural-Language Analysis*, CSLI, Stanford University.

OLAM – et semiautomatisk morfologisk og lydstrukturelt kodningssystem for dansk

Thomas O. Madsen, Hans Basbøll og Claus Lambertsen[1]
Syddansk Universitet, Institut for sprog og kommunikation
Campusvej 55, 5230 Odense M.
E-post: tmadsen@language.sdu.dk, hba@language.sdu.dk, claus@lambertsen.de

Abstract

This paper describes the OLAM system which is an integrated part of the Odense Language Acquisition Project. OLAM contains a database containing orthographic forms combined with morphological forms and information pertaining to segmentation and pronunciation. OLAM is first intended for semi-automatic coding of written text in Danish, second for complex linguistic searches and hypothesis testing.

Indledning

OLAM-systemet, der vil blive nærmere beskrevet i det følgende, udvikles i et samarbejde mellem Hans Basbøll, Thomas O. Madsen, begge fra Odense-Projektet i Sprogtilegnelse[2] (herefter OPS), Syddansk Universitet, og Claus Lambertsen, Berlin. Peter Molbæk Hansen, KUA, der er forfatter og copyrightindehaver til Gyldendals røde ordbog "Dansk Udtale" (herefter DU), 1990, deltager ikke i arbejdet med OLAM-systemet, men er orienteret om og har tilladt anvendelsen af DU i en elektronisk version med henblik på udviklingen af et morfologisk og lydstrukturelt søgnings- og kodningsprogram.

Hensigten med OLAM-systemet er på den ene side at udvikle en database, baseret på DU, indeholdende ortografiske former koblet til morfologiske former, segmenteringsoplysninger og udtaleformer. Denne database vil udgøre fundamentet for et kodningsprogram (OLAM-code), der skal anvendes til semi-

automatisk kodning af danske standard-ortografiske tekster, bl.a. et stort longitudinalt talesprogskorpus (Odense Tvillinge-Korpus) indsamlet af OPS og delvis transskriberet i løbet af projektperioden. På den anden side vil databasen blive udbygget med yderligere informationer og søgefaciliteter og på den måde udgøre en avanceret søgemaskine (OLAM-search) med mulighed for komplekse lingvistiske søgninger og testning af hypoteser.

OLAM er et projekt under udvikling. Mange af de beskrevne processer er enten helt eller delvist realiseret, men selve kodningsprogrammet forventes først at være operationelt i slutningen af 2001. Beskrivelsen i det følgende vil først og fremmest omhandle udviklingen af kodningsprogrammet OLAM-code, dvs. arbejdet med at omstrukturere databasen, at tilføje flere bøjningsformer, generere morfologiske kodningsformer m.v., og endelig opgaven med at indarbejde databasen i et operationelt kodningsprogram.

1. DU og OLAM

Udgangspunktet for arbejdet med OLAM-systemet, der som nævnt udvikles med henblik på et semi-automatisk morfologisk og lydstrukturelt kodningssystem for dansk standardortografi, er en elektronisk version af DU, som Claus Lambertsen for nogle år siden fik stillet til rådighed af Peter Molbæk Hansen. Claus Lambertsen har forarbejdet den elektroniske version af DU, så den senere kunne importeres i en FileMaker-database, som han har udviklet med henblik på bl.a. digraf- og difon-søgeprocedurer.

1.1. Strukturen i DU

Det vil altid være temmelig vilkårligt, hvor mange ord der medtages i en ordbog, uanset om det er en en- eller tosproget ordbog. At dansk er et sprog med vide muligheder for dannelse af sammensatte ord, gør ikke afgrænsningsproblematikken mindre relevant. Peter Molbæk Hansen pointerer i det indledende afsnit om 'Ordbogens principper og dens brug', at han i det konkrete udvalg af ord har

skelet stærkt til Retskrivningsordbogen (1986) "således forstået at der i udtaleordbogen ikke forekommer ret mange opslagsord der ikke også forekommer i Retskrivningsordbogen". Det skal dog pointeres, at det samlede antal opslagord i DU er noget mindre end i Retskrivningsordbogen, hvilket bl.a. skyldes, at DU ikke indeholder lige så mange sammensatte ord, da udtalen af langt de fleste sammensatte ord kan udledes af ordenes enkelte led, fx *kaffekande*[3]. Visse sammensatte substantiver er dog medtaget som opslagsord, fordi et eller flere af deres enkelte led ikke forekommer alene, og/eller deres udtale ikke kan konstrueres efter sammensætningsregler (fx *trykseksten*, *rødgrød*).

1.2. Opslagsord

DU indeholder 41102 ordbogsartikler, der er ordnet alfabetisk efter deres opslagsord. I forbindelse med import af den forarbejdede DU i en FileMaker database valgte vi at lade hvert kort indeholde ét opslagsord med tilhørende bøjningsformer. I en lang række tilfælde indeholder opslagsordet i DU en lodret streg ("|"), der deler ordet i en venstredel bestående af en "grafemisk konstant" (i GC-feltet[4]) og en højredel bestående af en "grafemisk endelse" (da endelserne endnu ikke er morfologisk identificerede, har vi valgt at betegne feltet "GV" for "grafemisk variabel").

Ved termen "grafemisk konstant" henviser vi til den del af et ord, der manifesteres ortografisk ens i alle ordets bøjningsformer, fx *cyk* i substantivet *cyk|-el*, *cyk|-(e)len*, *cyk|-ler* eller *vris* i adjektivet *vris|-sen*, *vris|-sent*, *vris|-ne*[5].

Den "grafemiske endelse" er den del af et ord, der ikke udgør en grafemisk konstant, men indeholder bøjningsspecifikke ortografiske oplysninger, fx endelsen *-pe* i infinitiv af verbet *hop|-pe*. Som det fremgår af eksemplet, er der ikke tale om morfologiske endelser, men grafemiske endelser indeholdende bl.a. bindekonsonanter.

Ved de bøjede ordklasser er opslagsordet på sædvanlig leksikografisk vis følgende: singularis ubestemt form af substantiver, singularis fælleskøn ubestemt

form af adjektiver, infinitiv ved verber. I forbindelse med hvert opslagsord er der desuden angivet en af følgende ordklasser: adj., adv., art., interj., konj., num., onom., pron., prop., præp., sb., vb.[6]

Endelig er DU struktureret således, at "homografe opslagsord", dvs. potentielle opslagsord, der manifesteres ortografisk ens, repræsenteres af samme opslagsord, hvis de har samme distributionsmønster i de for ordbogen relevante lydlige, ordklassemæssige og bøjningsmæssige henseender. Hvis de homografe opslagsord derimod har forskellige distributionsmønstre i en eller flere af disse henseender, repræsenteres de af forskellige artikler.

1.3. Bøjningsformer

Hvis vi ser bort fra de fuldt udskrevne særformer, der vil blive behandlet nedenfor, anføres kun den grafemiske endelse i andre bøjningsformer end opslagsformen, og en bindestreg ("-") repræsenterer her den grafemiske konstant, der er fælles for opslagsordets bøjningsformer.

Til opslagsordet følger i de fleste artikler et antal bøjningsformer. Følgende typer af bøjede former er almindeligvis angivet:

• Substantiver: bestemt form singularis og ubestemt form pluralis. Bestemt form pluralis er kun angivet i tilfælde af særformer (fx *børnene*, *gæssene*, se nedenfor) og i substantiver, hvis ubestemt form singularis er lig ubestemt form pluralis (fx *lam|-mene*, *kort|-ene*). Sidstnævnte kategori er desuden markeret i feltet PL=SG.

• Adjektiver: neutrum. Som følge af DUs (ortografibaserede) principper er der ikke anført en distinktion mellem adjektivers bestemte form og pluralis, hvilket betyder, at e-formen kun forekommer én gang. Komparativ og superlativ anføres, hvis adjektivet kan gradbøjes.

• Verber: præsens, præteritum, perfektum participium (og præsens participium ved verber, der ender på vokal) og imperativ. Imperativ, der altid er lig verbets

grafemiske konstant, markeres ved "-!" i feltet for den grafemiske endelse, fx GC: "*hop|*" og GV: "-!" for verbet *hop|-pe*.

• Pronominer: alle bøjede former.

Hvis der i forbindelse med et opslagsord eksisterer morfologiske dobbeltformer, fx flere præteritumsformer (*grinede*, *grinte*), er alle former medtaget, hvis de er forskellige fra alle øvrige former af ordet. Disse morfologiske dobbeltformer (MDF) har vi senere markeret i MDF-felter.

For de fleste ord er desuden medtaget den form, ordet optræder i som førsteled i sammensætninger. Hvis denne form ikke er medtaget, betyder det, at ordet – hvis det overhovedet kan optræde som førsteled – er sammenfaldende med opslagsformen i både ortografisk og lydlig henseende. Hvis førsteledsformen ortografisk, men ikke lydligt, er lig opslagsformen, er det markeret i feltet I SMS. Hvis førsteledsformen derimod ortografisk er lig hele opslagsformen (GC og GV), er den både markeret i feltet I SMS og GV-feltet er udfyldt af opslagsordets GV-indhold efterfulgt af en eller flere bindekonsonanter og et "+":

Opslagsord	GC	GV	eksempel
skab	*skab|*	*-s+*	*skabsdør*
føds| -el	*føds|*	*-els+*	*fødselshjælper*
dom|	*dom|*	*+*	*domhus*
	dom|	*-me+*	*dommedag*
	dom|	*-s+*	*domsmand*

Som det fremgår af sidste eksempel, kan et ord sagtens have flere former som førsteled i sammensætninger, fx *land|+, -e+, -s+; dag|+, -e+, -s+; bøn|+, -ne+*.

Endelig behandles visse bøjningsformer som særformer ("SF"). Hvis en bøjningsform er forskellig fra opslagsformen og ikke kan dannes ved en sammenkædning af den grafemiske konstant og en grafemisk endelse, udgør formen en

særform og hele formen er udskrevet, fx *var*, præteritumsformen af *vær|e*, *ænder*, pluralis ubestemt form af *and|*. Særformer indeholder ikke lodrette streger til adskillelse mellem en grafemisk konstant og grafemiske endelser, og indeholder altså heller ikke nogen grafemisk endelse. Vi har valgt at importere særformer i GC-felterne, men har samtidig markeret i feltet SF, at der er tale om særformer.

1.4. Udtaleformer

Til hver af de optagne ordformer er der angivet en eller flere udtaleformer. Lydskriften i DU udgør en tilpasset variant af den internationale standard IPA.

2. Identifikation og udspaltning af morfologiske endelser

Som det fremgår af beskrivelsen ovenfor, var den elektroniske version af DU, som vi fik adgang til, struktureret med henblik på at blive anvendt som udtaleordbog, dvs. ud fra et ortografisk princip og kun med ganske få morfologiske oplysninger. Vores første opgave var derfor at omstrukturere databasen fra primært at være organiseret efter ortografiske principper til at udgøre en morfologisk struktureret database.

Vores første problem i dette arbejde var, at der ikke eksisterede en en-til-en relation mellem felter og feltindhold, hvilket betød, at vi ikke kunne forudsige, hvilke bøjningsformer vi kunne finde i et specifikt (GV-) felt eller hvilke (GV-) felter en specifik bøjningskategori begrænsede sig til. Vi kunne med andre ord ikke blot flytte de morfologiske endelser til bøjningsspecifikke felter, men var nødsaget til at søge morfologiske bøjningsendelser i samtlige GV-felter.

For at løse dette problem blev der oprettet en række felter – ME-felter – der skulle indeholde specifikke identificerede morfologiske bøjningsendelser. For også at kunne segmentere komplekse morfologiske endelser, som fx pluralis bestemt form i *hus|-e-ne*, oprettede vi 2 ME-felter til hver bøjningsform, dvs. et ME01-felt og et ME02-felt pr. linje. Ganske vist kan der ud fra lingvistiske kri-

terier, fx Basbølls ordstrukturmodel baseret på endelsernes produktivitet (1998, 2001), argumenteres for, at der skal anvendes mere end blot to ME-felter pr. bøjningsform, men vi valgte at følge et minimalprincip[7]. Dernæst begyndte vi at søge og fraspalte de specifikke morfologiske bøjningsendelser, så enhver morfologisk endelse kunne placeres i et ME-felt, dog således at intet ME-felt kunne være udfyldt af mere end én morfologisk endelse[8].

I forbindelse med placeringen af morfologiske bøjningsendelser i hhv. ME01- og ME02-felter opererer vi med fire prioriterede generelle principper:

(1) hvis der i en grafemisk endelse kan identificeres to morfologiske bøjningsendelser, placeres den første i ME01 og den anden i ME02; (2) samme grammatiske kategori anbringes gennemført i enten ME01 eller ME02, uafhængigt af bøjningslinje; (3) Ø-endelser indsættes kun i ME-felter hvor andre udtryk for den pågældende morfologiske kategori ville stå; (4) hvis en grafemisk endelse kun indeholder én morfologisk bøjningsendelse (herunder "Ø"), placeres denne i ME01.

Vores overordnede strategi for segmentering af morfologiske endelser var at fraspalte maksimalt bagfra i GV-felterne, dvs. i de grafemiske endelser. Det forhold at opslagsordene er opdelt i en grafemisk konstant og en grafemisk endelse vha. en lodret streg betød, at vi var sikret mod at komme til at segmentere i den grafemiske stamme, fx at fraspalte *en* som ental bestemte form i *sten|*, *fænomen|*, *helgen|* eller *er* som præsens i *accepter|*, *administrer|*, *adopter|*.

Strategien med at fraspalte maksimalt bagfra i GV-felterne udtrykker en lingvistisk hypotese om, at segmentering af danske bøjningsendelser overvejende kan udføres vha. enkelte formmæssigt identificerbare morfologiske endelser (jf. Basbølls ordmodel, Basbøll 1998, 2001).

Nedenfor har vi opstillet nogle eksempler på søgnings- og fraspaltningsprocedurer samt placering af morfologiske bøjningsendelser i ME01- og ME02-felter.

Infinitiv er altid lig hele opslagsordet (GC og GV) i ordklassen verber og skal derfor kun søges i første linje (i felterne GC_01 og GV_01).

Hvis infinitivsendelsen ("e") overhovedet er til stede, er den altid placeret (sidst) i feltet GV_01, eventuelt efter en bindekonsonant (fx i *hop|p-e*), og kan uden videre flyttes til feltet A_ME01[9]. I tilfælde uden udtrykt infinitivsendelse indsættes et "Ø", fx i *gå|*.

I søgningen af præsens-endelsen "r" er vi pga. defektive former og morfologiske dobbeltformer nødsaget til at udvide søgefeltet til at omfatte alle GV-felter[10]. Alle identificerede præsens-endelser, dvs. "r", fundet ved søgning først på "*er" og derefter på "*r" kan herefter flyttes til feltet B_ME02. I tilfælde med "er" flyttes "e" til B_ME01, ellers indsættes "Ø" i B_ME01 (fx *hop|p-e-r*, *gå|-Ø-r*). Eventuelle bindekonsonanter bliver tilbage i GV-feltet (fx *p* i *hop|p-e-r*). Placeringen af "e" i B_ME01 skal opfattes som udtryk for, at basisformen for præsensformen *hopper* er en infinitivsform.

Principperne for segmentering af morfologiske bøjningsendelser beskrevet ovenfor vedrører udelukkende segmentering i GV-felterne. Når det drejer sig om særformer (markeret i SF-feltet), må vi gå lidt mere forsigtigt til værks, da særformerne ikke indeholder en lodret streg, der kan blokere for segmentering i den grafemiske stamme. Som overordnet strategi har vi valgt at udskrive alle særformer – først sorteret efter ordklasse og derpå efter deres "endelse" (dvs. om særformen ender på en bogstavsekvens, der også findes som fraspaltet bøjningsendelse). Med afsæt i denne liste afprøves, hvilke af søgningsprocedurerne fra GV-felterne, vi uden fejlanalyser kan overføre til særformerne.

3. Dannelse af nye bøjningsformer

Mht. bøjningsformer er udgangspunktet i DU, at bøjede former er udeladt, hvis deres lydskrift kan dannes ved blot at sammenkæde lydskriften af opslagsordet med lydskriften af en lydligt konstant endelse, eller hvis de i alle relevante henseender opfører sig parallelt med en medtagen bøjningsform. Med andre ord er bl.a. flere af de højfrekvente produktive endelser ikke medtaget, fx bestemt form

pluralis af substantiver (*-ne* i *bil|-er-ne*), præsens participium i de tilfælde, hvor verbet ender på en konsonant (*-ende* i *løb|-ende*), gerundium (*-en* i *handl|-en* og *-en* i *gå|-en*) og passiv (*-s* i *hop|p-ede-s*). For at OLAM-systemet kan fungere optimalt som et morfologisk og lydstrukturelt søgnings- og kodningsprogram, skal vi naturligvis indføje disse bøjnings- og udtaleformer i databasen.

I det følgende vil vi med udgangspunkt i verbers præsens participium og bestemt form pluralis af substantiver illustrere, hvilke procedurer vi tager i anvendelse for at danne de omtalte bøjnings- og udtaleformer.

Præsens participium er allerede udfyldt (i feltet F_ME01) for det mindretal af verber, der ender på en vokal. I alle andre tilfælde, dvs. hvor verbet ender på en konsonant, indsættes "ende" i F_ME01. Mht. udtalen kopieres infinitivens udtale og "nə" indsættes efter infinitivens "ə" (læs schwa).

Bestemt form pluralis af substantiver er ikke medtaget i normaltilfældet, dvs. hvis der kun er én ubestemt form i pluralis og bestemthedsformen er denne form efterfulgt af "*ne*". I de tilfælde, hvor feltet til bestemt form i pluralis (D_ME02) er tomt, kan vi indsætte den ortografiske form for ubestemt pluralis og tilføje "*ne*". Mht. udtalen indsættes udtaleformen af ubestemt form pluralis efterfulgt af " nə".

4. MOL-, SOL- og POL-former

Som det er beskrevet i indledningen, er formålet med OLAM-projektet bl.a. at udvikle et kodningsprogram til kodning af danske standard-ortografiske tekster[11]. Til det formål skal OLAM-databasen indeholde mindst tre typer informationer, nemlig morfologiske former, segmenteringsoplysninger og udtaleformer. De tre informationsniveauer betegnes inden for OLAM-systemet MOL-former (Morphological OLAM forms), SOL-former (Segmentized OLAM forms) og POL-former (Phonetic/Phonological OLAM forms).

Eftersom udtaleformerne, hvis 1. form udgør POL-formen, og SOL-formerne, der bygges op vha. grafemisk stamme, eventuelle bindekonsonanter og morfologisk(e) endelse(r), allerede er til stede i databasen, vil vi ikke komme nærmere ind på dem her. Procedurerne til generering af MOL-formerne vil derimod blive behandlet nedenfor.

Som det fremgår af eksemplet sidst i artiklen, N|tank-DEF:en, er en MOL-form typisk bygget op af følgende elementer (fra venstre mod højre): VERSALER, der angiver ordklasse (fx ADJ, N eller NUM[12]); symbolet "|", der adskiller ordklasseangivelse og opslagsordet; almindelige bogstaver, der angiver opslagsordet i standardortografi; et morfologisk grænsesymbol (uddybes nedenfor); VERSALER, der angiver den grammatiske kategori, efterfulgt af et kolon (fx DEF:, POSS:, PAST:); og endelig almindelige bogstaver, der angiver morfologisk(e) bøjningsendelse(r).

Grænsesymbolerne, der overvejende er morfologiske, anvendes på følgende måder:

Symbolet "-" er det umarkerede symbol før (adderet, dvs. ikke-fusioneret) morfologisk endelse i ME01-feltet (fx N|bil-PL:er), hvor "=" er det umarkerede symbol før morfologisk endelse i ME02-feltet (fx N|bil-DEF:en); begge symboler forekommer i N|bil-PL:er=DEF:ne.

Symbolet "&" findes før morfologisk endelse, der er delvist fusioneret, dvs. at der faktisk er udtrykt en segmenterbar endelse, men at den ikke er føjet til ordets normale stamme (fx *bøger*, N|bog&PL:er; *ænder*, N|and&PL:er)[13]; Symbolet "&" kan kun forekomme én gang i en MOL-form (aldrig sammen med "-"): *ænderne* N|and&PL:er=DEF:ne.

Symbolet "+" er placeret mellem leddene i sammensatte ord (fx *landmand*, N|land+N|mand);

"^" findes før "medfødte", totalt fusionerede, semantiske kategorier. Symbolet "^" er i realiteten ikke et morfologisk grænsesymbol, men indikerer en særlig relation mellem to indholdskategorier, nemlig "total fusion". Symbolet "^" an-

vendes i eksempler som *penge*, N|penge^PL, og *skib*, N|skib^NEUT, hvor det angiver, at *penge* er "medfødt" pluralis og *skib* er "medfødt" neutrum (markeret indholdskategori). Teknisk set er symbolet "^" af lavere rang end (de andre) morfologiske grænsesymboler, hvorfor operatoren implicerer, at de to omgivende elementer skal behandles som en enhed i relation til andre operatorer (hvad enten de omgivende elementer består af en stamme og en grammatisk kategori, fx *skib* og NEUT, eller to grammatiske kategorier, fx PL og DEF)[14]. Som illustration skal parenteserne i *musene* N|(mus-(PL^DEF:)ene) og *husene* N|(((hus^NEUT)-PL:e)-DEF:ne) opløses udefra og indad.

5. Kodning af tekster

Som det fremgår af ovenstående beskrivelse vil alle bøjningsformer, når MOL- og SOL-formerne er færdiggenereret, bl.a. bestå af en ortografisk form, en morfologisk form (MOL), en segmenteret form (SOL) og en udtaleform (POL). I forbindelse med kodning af danske standard-ortografiske tekster vil den ortografiske form kunne fungere som indgang til den specifikke bøjningsforms grammatiske og udtalemæssige former.

Det største problem forbundet med en "ord-til-ord"-identifikation af tekster består i det store antal homonymer, der eksisterer på dansk. Homografiproblemet udelukker desværre, at kodningsprogrammet kan foretage fuldautomatisk kodning af tekster, med mindre man indbygger en eller anden form for avanceret syntaktisk parser. Vi har valgt, i hvert fald i denne omgang, at løse homografiproblemet på den måde, at det er en operatør, der varetager opgaven med disambiguering i tilfælde af homografi.

I det færdige semiautomatiske kodningssystem vil resultatet af en søgning føre til en af følgende tre situationer:.

(1) kun én ortografisk form i databasen matcher inputformen. I dette tilfælde indsættes automatisk morfologisk, segmenterings- og udtaleinformation i de tre

følgende linjer under den identificerede inputform (hhv. i MOL-, SOL- og POL-linjen);

(2) ingen form i databasen matcher inputformen. I det tilfælde indsættes den ortografiske inputform (markeret med et særligt symbol) automatisk i tre følgende linjer under den søgte inputform;

(3) to eller flere ortografisk former i databasen matcher inputformen. I tilfælde af homografi skifter kodningsprogrammet til et layout, der anvendes i forbindelse med disambiguering. I disambigueringslayoutet har operatøren mulighed for at vælge den rette form ud fra oplysninger om MOL-, SOL- og POL-former. I eksemplet nedenfor vælger operatøren den rette af tre mulige MOL-, SOL- og POL-former.

	tanken	*tanken*	*tanken*
MOL-form	N|tanke-DEF:en	N|tank-DEF:en	N|tank-DEF:en
SOL-form	tanke|-n	tank|-en	tank-en
POL-form	tANgən	tAN'gən	ta:Ngən[115]

Som en ekstra sikkerhed har vi valgt, at operatøren skal godkende en hel ytring og dens kodning, før de tre kodningslinjer (MOL-, SOL- og POL-linjerne) overføres til det oprindelige dokument.

Efterord (juni 2002)

Pr. juni 2002 er status for OLAM-projektet følgende: En prototype af kodningsdelen er under fortsat afprøvning og OLAM-code er ved at blive implementeret. I løbet af sommeren 2002 foretages forskellige testninger og tilpasninger af systemet. I efteråret 2002 iværksættes den semiautomatiske kodning af vore tvillingedata (under projektet Danske Børns Sprogtilegnelse som støttes af Carlsbergfondet fra 2002). Parallelt hermed udvikles søgemaskinen OLAM-search

(under samme projekt). Der vil blive arbejdet henimod at etablere internetadgang til dele af systemet.

Noter:

[1] Rækkefølgen af forfatternavne udtrykker en vægtfordeling vedrørende udarbejdelsen af denne artikel, ikke OLAM-projektet som sådan.
[2] Odense-Projektet i Sprogtilegnelse (The Odense Language Acquisition Project) er et projekt støttet af Statens Humanistiske Forskningsråd og Syddansk Universitet, Odense Universitet, og består af 9 forskere fra Institut for Sprog og Kommunikation.
[3] Ordets udtale som førsteled i sammensætninger er dog angivet sammen med udtalen af andre bøjningsformer.
[4] Feltbetegnelser vil i det følgende være angivet med VERSALER, hvorimod feltindhold markeres ved dobbelte anførselstegn.
[5] Den lodrette streg ("|") er placeret så langt til højre i opslagsordet som muligt, for at der ved hver bøjningsform skal skrives så lidt som muligt.
[6] Her, som i resten af denne beskrivelse, anvender vi Peter Molbæk Hansens grammatiske betegnelser.
[7] Der findes bøjningsformer med tre morfologiske endelser, fx substantivernes pluralis bestemt form genitiv (*hus*| *-e-ne-s*), men da vi behandler genitiv som klitisk, skal vi ikke have et særligt ME-felt til genitiv.
[8] Det skal understreges, at antallet af morfologiske endelser her udelukkende vedrører udtrykskategorier, ikke antallet af (semantiske) indholdskategorier (fx udgør endelsen i *mus*|*-ene* kun én morfologisk endelse på udtrykssiden, men udtrykker på indholdssiden både pluralis og bestemthed).
[9] Af praktiske hensyn har vi valgt at referere til "linjer" i den oprindelige del af databasen, dvs. den grafemisk strukturerede del, med 01, 02, 03, osv., hvorimod "linjer" i den morfologisk strukturerede del af basen betegnes A, B, C, osv.
[10] Både grafemiske og morfologiske dobbeltformer håndteres ved procedurer, der af hensyn til artiklens omfang ikke vil blive beskrevet her.
[11] Med standardortografi henvises der til Retskrivningsordbogen 1986, da denne var udgangspunktet for udviklingen af DU mht. standardformer (jf. afsnit 1.1 Strukturen i DU).
[12] Grammatiske betegnelser og ordklassebetegnelser i det færdige OLAM-system afviger på en række punkter fra betegnelserne i DU, derimod svarer de overens med de (engelske) betegnelser, der anvendes inden for CHILDES (http://childes.psy.cmu.edu/).
[13] Distinktionen mellem "-" (ingen fusion) og "&" (delvis fusion) består i, hvorvidt der er en ortografisk forskel på opslagsordets stamme og den pågældende morfologiske forms stamme (eller mere præcist: et "&" markerer, at opslagsformens grafemiske konstant ikke er lig begyndelsen af den pågældende bøjningsforms grafemiske konstant et vice versa).
[14] Med andre ord kan de to indholdskategorier, der er forbundet af operatoren "^" (fx i "PL^DEF eller "hus^NEU"), ikke opløses i to selvstændige (ikke-tomme) elementer på udtrykssiden.
[15] Beklageligvis var det af tekniske årsager ikke muligt i nærværende artikel at gengive lydskrift med IPA-symboler. "A" svarer til fuldvokalen i *banke*, "N" svarer til sidste lyd i *streng*, "\" angiver schwa, "'" symboliserer stød, "a:" svarer til langvokalen i *gade*.

6. Litteratur

Basbøll, Hans. 1998. Nyt om stødet i moderne rigsdansk: om samspillet mellem lydstruktur og ordgrammatik. *Danske Studier* 1998:33-86.

BEST COPY AVAILABLE

Basbøll, Hans. 2001. Fuldproduktive bøjningsendelser og stød: nogle konsekvenser af en ny model for ordstruktur. In Pia Jarvad et al. (eds.) [Festskrift]. København:Hans Reitzels Forlag.

Dansk Sprognævn. 1986. *Retskrivningsordbogen*. København:Gyldendal.

Hansen, Peter Molbæk. 1990. *Dansk udtale*. København:Gyldendal.

Semantic Encoding of Danish Verbs in SIMPLE Adapting a verb-framed model to a satellite-framed language[1]

Bolette Sandford Pedersen & Sanni Nimb
Center for Sprogteknologi, Njalsgade 80,
DK-2300 S, Denmark
E-post: bolette@cst.ku.dk, sanni@cst.ku.dk

Abstract

In this paper we give an account of the representation of Danish verbs in the semantic lexicon model, SIMPLE. Danish is a satellite-framed language where prepositions and adverbial particles express what in many other languages form part of the meaning of the verb stem. This aspect of Danish – as well as of the other Scandinavian languages - challenges the borderlines of a universal, strictly modular framework which centralises around the governing word classes and their arguments. In particular, we look into the representation of phrasal verbs and we propose a classification into compositional and non-compositional phrasal verbs, respectively, and adopt a so-called *split late* strategy where non-compositional phrasal verbs are identified only at the semantic level of analysis.

1. Introduction

The aim of the EU-project SIMPLE (Semantic Information for Multifunctional Plurilingual Lexica) is to provide harmonised semantic lexicons for Natural Language Processing for 12 of the European languages. The project is an extension of the LE-PAROLE lexicons, which contain 20,000 entries with corresponding morphological and syntactic informa-

[1] This paper has also been published in Second International Conference on Language Resources and Evaluation , Athen, Grækenland 2000.

tion for each of the 12 languages that participated in the project (cf. Ruimy *et al.*, 1998).

The language specific encodings in SIMPLE are performed on the basis of a unified, ontology-based semantic model - the so-called SIMPLE model - representing an extended Qualia Structure based partly on Pustejovsky (1995), partly on experiences in preceding lexical projects such as Genelex, WordNet and EuroWordNet.

In this paper we focus on the problems encountered during the encoding of *Danish verbs* in SIMPLE, an encoding process which has been continuously supported by corpus data and where some principled solutions have been required in order to adapt the universal SIMPLE model to the empirical data of a Scandinavian language like Danish.

Speaking in Talmy terms (Talmy 1985), Danish is a typical satellite-framed language, meaning that prepositions and adverbial particles express what in many other languages form part of the meaning of the verb (cf. Harder, Heltoft & Thomsen 1996, Durst-Andersen & Herslund 1996, Herslund 1993 and Pedersen 1999). Thus, several of the most frequent verbs in Danish are relatively neutral with respect to semantic affiliation in the ontology as well as regarding event type; their affiliation being determined rather by the particle or the preposition than by the verb stem itself. In fact, from our corpus examinations we estimate that more than half of the verb senses relevant for SIMPLE (relevance is here solely based on frequency) is constituted by phrasal verbs which cannot be uniquely assigned a semantic type on the basis of the verb stem alone.

Representing this aspect in a lexicon is a challenge not only for traditional lexicography but even more for computational lexicography, which has a long tradition of a modular composition of the lexicon distinguishing strictly between morphology, syntax and semantics; and which is traditionally centralised around the governing word classes, nouns, adjectives

and verbs and the arguments that they take. Such a model seems intuitively better suited for a verb-framed language which encodes the core meaning components in the verbal stem.
Two questions need to be answered in order to propose a treatment of Danish phrasal verbs in the PAROLE/SIMPLE model:

Are phrasal verbs to be considered a morphological, a syntactic or a semantic phenomenon ?
How do we represent the semantics of phrasal verbs ?

In this paper, we recognise the fact that phrasal verbs challenge a strictly modular view of the lexicon and that they are an excellent example of why such a strict modularity is in essence not ideal for the intuitive treatment of a satellite-framed language like Danish. The compromise that we suggest to handle this fact is a so-called *split late* strategy where phrasal verbs are only represented as such at the semantic level irrespective of whether they are compositional or non-compositional in meaning.
This has as consequence a somewhat controversial analysis at the syntactic level, where particles that are in fact incorporated in the verb receive a treatment parallel to weakly bound prepositional objects. In the following we first give an introduction to the event ontology in SIMPLE (Section 2); then we proceed to a description and classification of Danish phrasal verbs (Section 3), and finally we propose a treatment of these in the modular PAROLE/SIMPLE model (Section 4).

2. Verb encoding in SIMPLE

One of the fundamental assumptions behind the SIMPLE model is that word senses differ in terms of their internal complexity and that this com-

plexity can be described on the basis of an ontology established along different dimensions (Lenci *et al.* 2000). Some word senses can be described by means of *simple* types, which means that they inherit their information from only one mother node in the ontology; others are more complex and thus inherit information from several mother nodes following the principle of orthogonal inheritance[1].

These multiple dimensions of meaning are represented in SIMPLE by means of an extended Qualia Structure model based on Pustejovsky (1995) encompassing a set of semantic relations such as *is_a*, *part_of*, *has_as_parts*, *resulting state* etc. for each qualia. Furthermore, regular polysemous classes are represented in SIMPLE via the additional type: *complex* which establishes a link between systematically related senses. In the case of verbs, complex types are in particular applied for causative/decausative alternations, as in *jeg triller bolden* (I roll the ball) vs. *bolden ruller* (the ball rolls).

In the SIMPLE ontology, verbs and event nouns are affiliated under the node *event* which again dominates a whole sub-hierarchy of types to be used when classifying different kinds of events (cf. Lenci *et al.* 2000:pp. 29-30). The ontology for events is influenced by several sources including in particular WordNet (Miller *et al.* 1990), EuroWordNet (Alonge *et al.* 1998) and Levin verb classes (Levin 1993). One of the aims has been to find a number of event classes which is richer than that of WordNet comprising 15 classes and less detailed than Levin's 234 classes. Thus, the SIMPLE event ontology comprises 59 classes grouped into 7 core categories as seen in Figure 1 (next page).

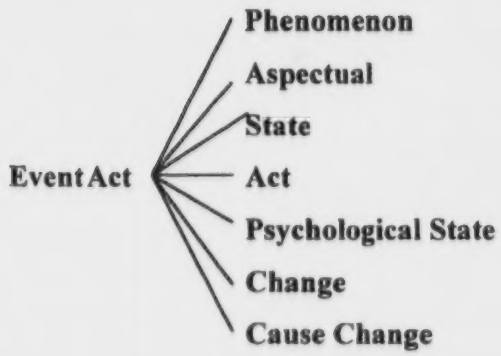


Figure 1: Core event types in SIMPLE

Three fundamental aspects have been considered in the classification: event type, i.e. basically whether a verb sense denotes a state, an act or a transition argument structure; i.e. arity and type of arguments subcategorised for by the verb sense causativity; i.e. whether a verb sense is causative or non-causative; the former always being represented by a unified type.

When a verb is described in SIMPLE, the appropriate set of senses to be assigned to the verb is first established, preferably on the basis of other dictionary sources as well as on corpus examination. Each sense constitutes what is labelled a semantic unit (a *SemU*) which is then assigned a semantic type according to the ontology. Each SemU is further linked to its corresponding syntactic and morphological units. Consequently, the model permits to distinguish different syntactic behaviours on pure syntactic criteria and independently of whether they share the same meaning or not (see Ruimy *et al* 1998). Figure 2 illustrates the linking of units at the different levels (*Mus* = morphological unit, *SynU* = syntactic unit, *Semu* = semantic unit).

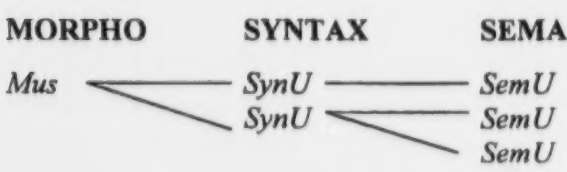


Figure 2: Linking of units at the three levels

As can be seen, one syntactic unit can very well link with two semantic units which are then maybe assigned two different types in the event ontology.

As regards the internal structure of the semantic unit; consider Figure 3 below which gives the contents of the semantic unit of one of the senses of *krydse* (cross) in the Danish SIMPLE lexicon:

Semantic Unit	*krydse_CHL* (cross)
Definition:	*Bevæge sig tværs over et åbent område* (Nudansk Ordbog) (move across an open area)
Corpus example:	*Drengen krydsede sporet ved stationen, men så ikke toget* 'The boy crossed the rails at the station but he didn't see the train'
Semantic type:	Change of location
Unification Path:	Change/Agentive
Domain:	General
Argument Structure	ARG1 ARGDirection
Selectional Restrictions	ARG1= Human OR Animal OR Vehicle Direction = Concrete
EventType	Transition

Formal quale:	Is_a = *ændring* (change)
Agentive quale:	Agentive = *bevæge_sig* (move)
Telic quale:	Nil
Constitutive quale:	Resulting_State = *være* (be) Direction= forwards
Systematic Polysemy	Nil
Synonymy	Nil

Figure 3: The semantic unit for *krydse* (cross)

The first slot 'Semantic Unit' refers to the word described; in this case *krydse*. The suffix CHL refers to the semantic type 'Change of location' and is practical in order to keep track of other eventual readings of the same word which could require multiple linking from syntax.

The next slot, 'Definition', is preferably taken from a Danish medium-sized dictionary and helps define the actual sense. 'Corpus example' is taken from a Danish collection of corpora (Berlingske Korpus of 20 mill. words and Bergenholtz Korpus of 4 mill. words) and is also meant as a help to the user. The example chosen should be typical for the sense and should exemplify argument realisation and typical selectional restrictions. 'Semantic type' refers to the concept in the ontology 'Change of location' which is placed as a subtype to 'Change'. 'Unification Path' gives the unification path for the unified type; in this case the type inherits from both 'Change' and 'Agentive'. 'Argument structure' should be self-explanatory; however, it should be noted that each language group has here been relatively free regarding how to analyse predicates. The Danish lexicon is based on the linguistic specifications developed within an EU-grammar project (LINDA – Linguistic Specifications for Danish (cf. Underwood et al. in press)).

As regards 'Selectional restrictions', the concepts of the ontology are applied; thus 'Animal', 'Human' and 'Vehicle' are concepts of the ontology applied for concrete nouns. 'Event type' can be either 'state', 'process' or 'transition' and is meant to refer to the 'neutral' interpretation of the verb in question – a somewhat problematic issue for Danish which we shall come back to in Section 3. In the case of *krydse*, however, there are no problems in assigning the value 'transition'.

Event type is followed by the four Qualia Roles, (i) the formal role, which provides information that distinguishes an entity within a larger set (*krydse* 'Is_a' *ændring* (cross 'Is_a' change), (ii) the agentive role, which concerns the origin of an entity (in this case *bevæge_sig* (move)) (iii) the telic role, which concerns the typical function of an entity (*krydse* has no such function), and finally (iv) the constitutive role, which expresses a variety of relations concerning the internal constitution of an entity (in this case 'Resulting State' which is to be another place (*være)* and 'Direction' 'forwards'). 'Systematic polysemy' and 'Synonymy' relations are not relevant for the encoding of *krydse.*

3. Danish verbs and adverbial particles

3.1. Unit accentuation as a linguistic test

When analysing Danish verbs and their satellites, unit accentuation proves to play a central role (see Scheuer 1995 and Harder, Heltoft & Thomsen 1996). In fact, unit accentuation can be used as a linguistic test in order to distinguish phrasal verbs from other verbs combined with adverbial particles. Consider the examples below:

(1) *Han 'blev 'væk*
'he stayed away'

(2) *Han blev 'væk*
'he got lost'

Example (1) has stress on both the verb and the particle indicating the fact that we are dealing with a simplex verb *blive* (stay) combined with an adverbial modifier *væk* (away).
In contrast, in (2) absence of full stress on the verb indicates that the verb does not constitute a clausal predicate on its own but that the element that receives full stress (the particle) should be interpreted as part of the semantics of the predicate. In the case of (1), *blive* can be described as a state verb - i.e. as non-transitional - subcategorising for a locational argument; in the case of (2), we must consider *blive væk* as one semantic unit; i.e. a phrasal verb of the type transitional with a 'change of location' assignment.

3.2. Compositional vs. non-compositional phrasal verbs

When looking closer into the category of verbs which can be categorised as phrasal verbs according to the test described above, a blurred picture emerges between those phrasal verbs that are compositional in meaning and those that are not. With compositionality we understand that both the host verb and the particle retain their core meaning as is normally the case when directional particles are combined with motion verbs as in:

(3) *han løb 'ud*
'he ran out'

(4) *han gik 'op*
'he walked up'

Compositionality is an important parameter when deciding how to represent a phrasal verb in the computational lexicon, and actually; in the tra-

ditional lexicon this distinction is also usually maintained although several Danish dictionaries are not completely clear on this point. Normally, however, only the non-compositional phrasal verbs find their way into a traditional lexicon as *sublemma* as for instance in the case of *vaske op*:

(5) *han vaskede 'op*
he washed up
'he did the dishes'

whereas the compositional phrasal verbs, which are predictable in meaning and often productive wrt. the directional particle to be connected with, are rather described by means of valency patterns in the 'core' entry, as in *løbe op/ned/ud.* ('he ran up/down/out..) resulting in the following valency pattern description: SUBJECT+DIRECTIONAL[2].

It should be noted that such an analysis may coincide with the description given for some non-phrasal verbs, resulting in the confusing fact that predictable phrasal verbs and non-phrasal verbs combined with directional particles are described in the same fashion in the lexicon. Consider the example below, which cannot be considered a phrasal verb according to the accentuation test, and where *ned* is thus not incorporated in the verb:

(6) *han 'kiggede 'ned*
'he looked down'

Such an example would usually be treated similarly to examples like (3) and (4) above.

3.3. Motion verbs as a special semantic class

The best way to solve this apparent contradiction, is to recognise that motion verbs constitute a very special semantic class. Motion verbs oc-

curring as phrasal verbs are mostly predictable in meaning since the directional particle is usually to be understood in its core sense; motion verbs thus constitute a unique semantic class by admitting the directional marker to be understood as *incorporated* in the verb as is shown by the accentuation test. This relates well to the fact that other motion verbs inherently express direction without the need for a directional particle; (cf. *krydse* (cross) in Section 2). It also opens for the possibility of viewing the expression of *direction* as a regular syntactic pattern which alternates with directional prepositional phrases:

(7) *han løb ud / til bageren / hen til skolen*
'he ran out/ to the Baker's / over to the school'

In the group of motion verbs that can occur as compositional phrasal verbs should also be included 'Cause change of location' verbs such as *flytte* (move), which have the same characteristics:

(8) *han flyttede ud/ ind/ hen til skolen*
'he moved out/ in/ over to the school

Furthermore, this group includes verbs from other domains that are 'coerced' into motion verbs when a directional particle is added:

han humpede hjem på sit forslåede ben
'he limped home on his battered leg'
han skramlede ned ad vejen i den gamle bil
'he rattled down the road in his old car'

3.4. Particles and prepositions acting as aspectual markers

It is a well-known characteristic of most satellite-framed languages that the particle acts as an aspectual marker typically changing a process verb

into a transition verb as seen in examples (3) and (4), where process motion verbs are changed into becoming change of location verbs. In some cases the aspectual marking is the only function of the particle as in the examples below where process verbs like *spise* (eat) and *drikke* (drink) are changed into transition verbs:

(11) *han spiste 'op* TRANSITION
'he ate up'

(12) *han drak 'ud* TRANSITION
he drank out
'he drank up'

But there also exist directional particles that are *not* transitional; in the examples below the meaning is that of a process even if a directional particle is added:

(13) *han gik hjemad* PROCESS
'he walked homewards'

(14) *han daskede rundt* PROCESS
'he sauntered about'

Likewise if two directional particles with opposite meaning are applied together as in:

(15) *pendulet svingede frem og tilbage* PROCESS
'the pendulum swung forwards and backwards'

Note also that the preposition *på* (on) can have an aspectual marker function (marking a process rather than a transition) when combined with a verb like *bygge* (build) which normally denotes transitions when combined with a nominal phrase:

(16)a. *hun byggede et hus* TRANSITION
'she built a house'

(16)b. *hun byggede på et hus* PROCESS
she built on a house
'she was building on a house'

Several other features can also influence the event type, such as definiteness of the object, addition of a resulting state etc.

3.5. Concluding remarks

In the preceding sections we have identified three kinds of constructions involving directional particles:

- simplex verbs combined with adverbial modifiers ((1), (6))
- phrasal verbs which are compositional, i.e. predictable in meaning ((3)(4))
- phrasal verbs which are non-compositional, i.e. idiosyncratic ((2), (5))

We have also seen that the second group of verbs is constituted by motion verbs; a unique semantic class in the sense that it admits the directional marker to be understood as *incorporated* in the verb in spite of the fact that the verb itself contributes to the meaning of the expression. Finally, we have looked at particles and at the preposition *på* as aspectual markers which switch the event type of a verb into another type.

4. Representing phrasal verbs in the modular PAROLE/ SIMPLE framework

4.1. Lemma identification and *split late* philosophy in PAROLE/ SIMPLE

The question is now how to represent these aspects in a verb-framed model like the SIMPLE event ontology where one of the basic underlying classification criteria has been exactly the event type. Leaving the simplex verbs aside in this discussion, it seems obvious that the idiosyncratic phrasal verbs must be fully lexicalised at the semantic level since their meaning is unpredictable and therefore requires a semantic description of its own. For the compositional phrasal verbs on the other hand, we can opt for either a fully lexicalised representation or for a directional slot representation of some kind. In any case we need to consider more thoroughly the whole PAROLE/SIMPLE structure in order to decide for a convenient strategy since the representation of the *lemma* as such, as well as the representation of syntactic information, prove to be highly relevant for this discussion.

When deciding how to represent the semantics of phrasal verbs in PAROLE/SIMPLE, we need not only consider how to represent them at the semantic level, but also to find a principled solution wrt. their representation at earlier levels; i.e. the morphological and the syntactic level. An interesting aspect of the PAROLE/SIMPLE model is that there exists no *lexical* unit or *lemma* as such in the traditional sense of the word. In order to identify what in traditional lexicography is conceived as a lemma, one has to start from the semantic unit and work back through the syntactic and morphological levels and gather all the relevant information.

This is due to the fact that the Danish PAROLE/SIMPLE lexicon consistently applies the so-called *split late* strategy. A split late strategy implies

that only what can be identified at a particular level of analysis as two different units (morphology, syntax or semantics) should result in the entry being split into separate units. Thus, for the two homographs of *love* (promise, praise) for instance, we find the following representation in the Danish PAROLE/SIMPLE lexicon:

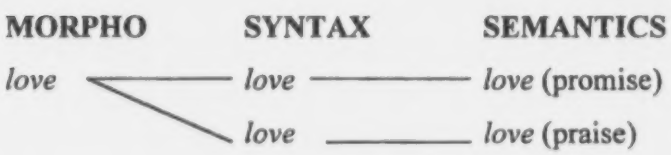


Figure 4: The representation of *love* (promise, praise)

In other words, even if we speak of homographs with different etymology and completely unrelated meanings, only one representation is given at the morphological level since the two are identical from a purely morphological point of view. The split into two units is realised at the syntactic level since the valency patterns of the two verbs differ; the former being ditransitive and the latter transitive.

For the representation of phrasal verbs several approaches can be adopted. We can either lexicalise a phrasal verb like *vaske op* at the morphological level and thus treat it as a completely different lexeme than *vaske*:

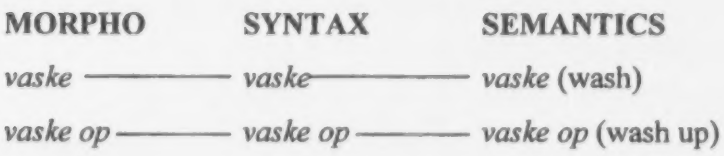


Figure 5: 'Split early' representations of *vaske* and *vaske op* (wash, do the dishes)

Or we can split at the syntactic level and lexicalise the phrasal verb *vaske op* at this level:

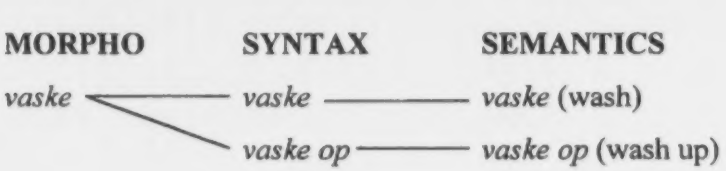


Figure 6: Splitting in syntax of *vaske* and *vaske op* (wash, do the dishes)

It does, however, match the split late strategy better to make the distinction at the semantic level and let the particle be treated as an optional complement at the syntactic level:

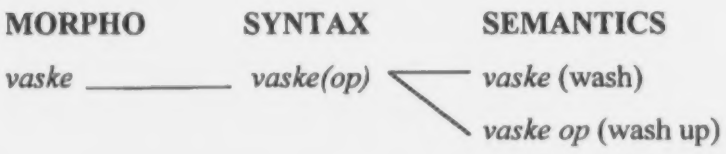


Figure 7: 'Split late' representations of *vaske* and *vaske op(*wash, do the dishes)

To consider *op* an optional complement to *vaske* may be controversial from a syntactic point of view but it has the advantage of leaving the decision of whether we are dealing with a compositional or a non-compositional phrasal verb for the semantic level where it actually belongs since it is basically a semantic distinction. Especially in cases of *ambiguity* (i.e. where both a compositional and a non-compositional interpretation is possible, as in *gå op* which can either be compositional meaning 'walk upwards' or non-compositional meaning 'cancel out') this is convenient since it prevents unnecessary overgeneration at earlier levels and allows for a unified syntactic description of directionals at the

syntactic level irrespective of whether these are expressed as particles or as prepositional phrases3:

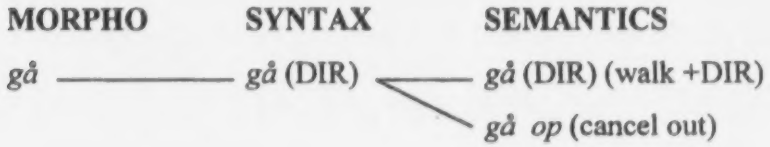


Figure 8: 'Split late' representation of *gå op* (walk upwards, cancel out)

4.2 The semantic unit

The non-compositional phrasal verb *gå op* (cancel out) is given a particular, fully 'lexicalised' representation in the semantics and is assigned the type 'Identificational State' from the ontology denoting states which ascribe a property to an entity and e.g. identity and resemblance with another entity:

Semantic unit:	*gå op_IDS* (cancel out))
Definition:	*(om regnestykke og kabale) løses så der ikke bliver nogen rest* (Nudansk Ordbog) '(about calculations and patiences) solve so that there is no remainder'
Corpus example:	*..men for at få regnestykket til at gå op måtte han inddrage naboens grund* 'but in order to make the calculation cancel out he had to include the neighbour's garden'
Semantic type:	Identificational state
Sem. Supertype:	Relational state
Event type:	State
Domain:	Mathematics
Predicative repres.:	ARG1
Selectional restrictions:	ARG1= Representational
Formal quale:	isa = *relation* (relation)
Agentive quale:	Nil
Telic quale:	Nil
Constitutive quale:	Relates = numbers
Systematic Polysemy:	Nil
Synonymy:	Nil

Figure 9: Semantic representation of *gå op* (cancel out)

As for the compositional phrasal verbs, the representation in Figure 8 has the advantage that these are treated in the same way as constructions with weakly bound preprositions (see example 7). Such a generalisation opens

for a representation at the semantic level where only one semantic unit is established for *gå* (walk) as depicted below:

Semantic unit:	*gå_MOV* (walk - move reading)
Definition:	*komme frem ved at sætte den ene fod foran den anden* (Nudansk Ordbog) 'proceed by putting one foot in front of the other'
Corpus example:	*Vi skal* ***gå*** *hen til telefaxen , vente mens den kalder op osv.* 'we have to **walk** over to the fax machine, wait while it makes the call etc.'
Semantic type:	Move
Sem. Supertype:	Act
Event type:	Process
Domain:	General
Predicative repres.:	ARG1 (DIR)
Selectional restrictions:	ARG1= Human OR animal DIR= Concrete
Formal quale:	isa = *bevæge sig* (move)
Agentive quale:	Nil
Telic quale:	Nil
Constitutive quale:	Manner = yes
Systematic Polysemy:	Nil
Synonymy:	Nil

Figure 10: Semantic representation of *gå* (walk)

One semantic unit can now account for *gå* in phrases like *han gik 2 km, han gik ud, han gik op, han gik hen til telefaxen* etc. (he walked 2 km, he walked out, he walked up, he walked over to the fax machine) since no lexicalisations of the phrasal verbs are involved. The problem of event structure assignment, is however *not* solved by this approach. In the SIMPLE Specifications (Lenci *et al.* 2000) it is stated that event type is meant to abstract away from the possible effects by complements and adjuncts – therefore the semantic unit in Figure 10 is assigned the event type 'process'. Thus, in this representation we ignore the fact that *gå* in some cases denotes a process and in other cases a transition and that there holds a regular pattern between the two that could have been represented as systematic polysemy between 'move' constructions and 'change of location' constructions. Again, the modular organisation of the SIMPLE model centralised around the governing word classes is not completely well-suited for this kind of phenomenon.

In the present representation it is thus up to a particular grammar or lexical rule set to capture this pattern which – as is well-known under the term *path-manner divergences* (cf. Talmy 1985, Slobin 1996, Pedersen in press and many others) - is crucial for instance in case of translation between Germanic and Romance languages. This is caused by the fact that Romance languages usually lexicalise the direction component (or 'path' in Talmy's terms) in the verb stem whereas Germanic languages lexicalise the manner component in the verb stem. So, *gå i to timer* (walk for two hours) would in Spanish translate into *caminar durante dos horas* whereas *gå ud* (walk out) would translate into a 'Change of location' verb: *salir* (exit). In other words, in Spanish the two verbs *caminar* and *salir* are placed in two different parts of the ontology; under 'Move' and 'Change of location', respectively, whereas for Danish we would only have one semantic unit affiliated under 'Move'.

5. Conclusions

Strict modularity as well as centering around the governing word classes and their arguments is an obvious approach for a computational, multi-purpose lexicon where the idea is that the lexicon should be usable for different kinds of NLP applications requiring different levels of linguistic information.

In this paper, we have discussed some of the linguistic problems encountered when adapting such a model to Danish verbs, and in particular to Danish phrasal verbs, which are not easily described from a modular point of view since they incorporate particles and therefore are discontinuous. In other words their analysis lies in the borderline between morphology and syntax on the one hand, and syntax and semantics on the other. In order to overcome this problem we have suggested a split late approach where at the morphological and syntactic levels simplex verbs with directionals, compositional phrasal verbs as well as non-compositional phrasal verbs are treated alike: the particle is treated as a valency slot filler even if from a semantic point of view it is incorporated in the verb.

This approach leaves it for the semantic level to differentiate between the different kinds of particle constructions since it is at this level the proper disambiguation can take place between the ones that are non-predictable in meaning and should therefore be lexicalised and those that are predictable in meaning and can therefore be described in a regular fashion.

Attempts to harmonise linguistic descriptions of different European languages into a universal model constitutes a challenging task but they also bring linguistic research further. Thus, the scope of the SIMPLE project makes it a truly pioneering project for Danish and considering the current status of language technology for the 'small' European languages, the development of these harmonised large-scale semantic lexicons is a first

step in the right direction for creating advanced language technology also for less widely spoken European languages.

Noter:

1 By 'orthogonal inheritance' we understand multiple inheritance with the restriction that a feature can only inherit its value from *one* mother node from the same partition. Thus, in SIMPLE each meaning dimension (each qualia role) establishes its own partition.
2 We must remark here, however, that frequency also plays a role in the construction of most modern lexica; thus very frequent phrasal verbs do tend to figure as sublemma even if they are predictable in meaning.
3 Two aspects should be noted here: firstly, the strategy proposed here obviously introduces an element of non-compositionality between syntax and semantics; secondly, for a lexicon meant for speech recognition we would probably prefer a *split early* strategy (at the morpohological level) since ambiguity would then be eliminated by information on stress.

6. References

Alonge, A., N. Calzolari, P. Vossen, L. Bloksma, I. Castellon, M.M. Marti, W. Peters. 1998. The Linguistic Design of the EuroWordNet Database. In P. Vossen (ed.) *EuroWordNet, A Multilingual Database with Lexical Semantic Networks*. Dordrecht, Boston, London:Kluwer Academic Press

Braasch, A. & B. Pedersen. 1999. En stor sprogteknologisk ordbog for dansk - med særligt fokus på håndtering af flertydighed i en niveaudelt ordbog. In *7. Møde om Udforskning af Dansk Sprog*. Århus:Århus Universitet.

Durst-Andersen, P. & M. Herslund 1996. The syntax of Danish verbs: Lexical and syntactic transitivity. In: E. Engberg-Pedersen et al. (eds.) *Content, Expression and Structure. Studies in Danish Functional Grammar*. Amsterdam: John Benjamins.

Harder, P., L. Heltoft & O.N.Thomsen. 1996. Danish directional adverbs, content syntax and complex predicates: A case for host and co-predicates.

In: E. Engberg-Pedersen et al. (eds.) *Content, Expression and Structure. Studies in Danish Functional Grammar*. Amsterdam: John Benjamins.

Herslund, M. 1993. Transitivity and the Danish Verbs.In *LAMBDA no. 18,* Københavns:Handelshøjskolen i København.

Lenci, A. F. Busa, N. Ruimy, E. Gola, M. Monachini, N. Calzolari, A. Zampolli, J. Pustejovsky, E. Guimier, G. Recourcé, L. Humphreys, U. Von Rekovsky, A. Ogonowski, C. McCauley, W. Peters, I. Peters, R. Gaizauskas, M. Villegas, O. Norling-Christensen. 2000. *SIMPLE Linguistic Specifications*. Pisa:University of Pisa.

Levin, B. 1993. *English Verb Classes and Alternations, A Preliminary Investigation*. Chicago and London:The University of Chicago Press.

Miller G., R. Beckwith, C. Fellbaum, D. Gross, K.J. Miller. 1990. An Online Lexical Database. *International Journal of Lexicography,* 3(4):235-244.

Pedersen, B.S. 1999. Systematic Verb Polysemy in MT: A Study of Danish Motion Verbs with Comparisons to Spanish. In H. Somers (ed.): *Machine Translation* 14(1):39-86. Dordrecht:Kluwer Academic Publishers.

Pustejovsky, J. 1995. *The Generative Lexicon*. Cambridge, MA:The MIT Press.

Ruimy, N. O. Corazzari, E. Gola, A. Spanu, N. Calzolari, A. Zampolli. 1998. The European LE-PAROLE Project: The Italian Syntactic Lexicon. In: *First International Conference on Language Resources & Evaluation*. Granada.

Scheuer, J. 1995. *Tryk på danske verber*. RASK Supplement (4). Odense:Odense Universitetesforlag.

Slobin, D. 1996. Typology and Rethoric: Verbs of Motion in English and Spanish. In: Shibani and Thomsen (eds.) *Grammatical Constructions: Their Forms and Meanings*. Oxford:Oxford University Press.

Talmy, L. 1985. Lexicalisation Patterns: Semantic Structures in Lexical Forms. In T. Shopen (ed.) *Language typology and syntactic description* (Vol. 3): *Grammatical Categories and the Lexicon*.Chicago:Press Syndicate of the University of Chicago.

Underwood, N., C. Poulsen, P. Paggio, A. Neville, B.S. Pedersen, B. Ørsnes, A. Braasch (forthcoming). LINDA - Linguistic Specifications for Danish. In: *CST Working Papers*. København:Center for Sprogteknologi.

En analyse af realisationer af fonemet [A:] i udvalgte regioner og aldersgrupper i SpeechDat 2

Jacob Teckemeier og Kim Hornbech
Aalborg Universitet
E-post: j_teckemeier@hotmail.com, k_hornbech@hotmail.com

Abstract

In our BA dissertation, we dealt with the question whether regional pronunciation variations of the phoneme [A:][1] could be detected by means of an F´-analysis in the Danish version of the speech data base SpeechDat 2. We chose to analyse limited segments of SpeechDat 2 which were selected under consideration of gender, age and region. F´analysis was run on the segments and we additionally examined the results using various statistic implements. Our investigation displayed patterns in the pronunciation which however were so vague that they could merely be described as tendencies. Upon this conclusion we proposed certain methodical adjustments as well as elaboration on the speech data base SpeechDat 2.

Indledning

Nærværende artikel er et sammendrag af vores bachelorprojekt, der dannede baggrund for vores oplæg til DALF´s årsmøde 2000. Formålet med vores projekt var ved hjælp af en F´-analyse at fastslå, hvorvidt det var muligt at efterspore tendenser i udtalen af fonemet [A:] i udvalgte regioner og aldersgrupper i taledatabasen SpeechDat 2. At projektet lod sig gennemføre, kan i høj grad tilskrives den store imødekommenhed Center for Personkommunikation udviste over for vores ide. Center for Personkommunikation stillede således undersøgel-

sesmaterialet SpeechDat 2 til vores fulde disposition og var desuden behjælpelig med implementeringen af de tekniske dele af analysen.

1. Materiale

1.1. SpeechDat 2

Taledatabasen SpeechDat 2 er blevet til på et initiativ fra EU og skal anvendes til at udvikle stemmestyrede servicefunktioner over det faste telefonnet.
Den danske version af SpeechDat 2 er den største taledatabase af sin art, da dens korpus er indtalt af 4000 talere der hver har bidraget med 43 ytringer. Registreringen af materialet foregik ved, at talerne ringede ind og oplæste ytringerne fra et fortrykt ark. Der forekom dog også spontane ytringer, da talerne f.eks. skulle oplyse deres geografiske placering. Bidragene blev elektronisk registreret og efterfølgende kontrolleret for at sikre en ensartet kvalitet i hele databasen. Under opbyggelsen af SpeechDat 2 har en række strenge kriterier været medvirkende til at sikre, at databasen fik nøjagtig det udseende og indhold, man ønskede. Det var således krav om ligelig kønsfordeling, ligesom antallet af medvirkende talere var i proportionelt forhold til befolkningstallet i de regioner, man ud fra dialektale hensyn havde inddelt landet i.

2 Metode

2.1. Begrundelse for [A:]

Når vi valgte kun at beskæftige os med et enkelt fonem, skyldtes det, at vi dels var nødt til at forholde os til grænserne for omfanget af et bachelorprojekt, dels at vi med det fonem, vi havde valgt at undersøge kunne have god grund til at tro, at det kunne reflektere eventuelle udtaleforskelle.

---

[1] In this paper,43 we make use of the Danish phonetic alphabet Sampa.

Valget af [A:] var motiveret af den fluktuerende udtale a-lydene traditionelt har haft på dansk. Otto Jespersen var en af det første til at påpege en tendens til at den korte a-lyd blev fremskudt. Frem mod århundredskiftet blev denne fremskydning mere og mere udbredt for sluttelig at blive den dominerende udtale. Der var dog flere undtagelser hvor den lange a-lyd anvendtes foran alveolærlyde og nul for eksempel i ord som "datter" og "alter".

Den typiske lange realisation af bogstavet *a* var i begyndelsen af det nittende århundrede den tilbagetrukne og åbne *a*-lyd. Denne realisation var dominerende indtil 1875, hvorefter tendensen blev, at lyden blev fremskudt og lukket. Imidlertid skete der også en modsatrettet udvikling, idet det tilbagetrukne og åbne [a:] hvor det efterfølges af et [R] smeltede sammen med dette og blev til [A:]. Resultatet af dette kan i dag spores i far, skare og spar.

Da den sproglige udvikling naturligvis sker i forskellige tempi og varierer fra landsdel til landsdel, var det vores antagelse at man ved en F'-analyse måtte kunne klarlægge eventuelle mønstre i variationen af udtalen af [A:].

(Brink, Lars, et.al., 1974)

2.2. Udvælgelse af undersøgelsesgrupper

Det første udvælgelseskriterium, vi blev stillet over for, var, om undersøgelsen skulle omfatte begge køn. Imidlertid valgte vi at koncentrere os om mandestemmerne, da den tekniske analyse viste, at disse ud fra en akustisk vurdering viste sig at være mere velegnede end kvindestemmerne.

Da de mandlige deltagere aldersmæssigt spredte sig fra børn til oldinge var vi nødsaget til endvidere at afgøre, hvorvidt vi ønskede at inddrage hele aldersspændet i vores undersøgelser. Vi endte med at lægge os fast på tre aldersgrupper, der byggede på den inddeling, der var foretaget i databasen. De tre grupper bestod af 16-30-årige, 31-45-årige og 46-60-årige. Årsagen til, at vi satte den aldersmæssige grænse ved 16 år, var at personer under 16 endnu ikke har fuldt

udviklede taleorganer, hvorfor den lydlige kvalitet kan være svingende. Endvidere satte vi den øvre grænse ved 60 år, da ældres taleorganer ændres hvorfor den lydlige kvalitet igen kan være varierende.

Det sidste kriterium, vi opstillede, var af geografisk og dermed dialektal karakter. I SpeechDat 2 er landet inddelt i elleve regioner, men af hensyn til det maksimale omfang af opgaven kunne en undersøgelse af alle disse ikke udføres. Vi begrænsede analysen til at omfatte tre af de regioner, der var afstukket i databasen. Det blev regionerne Nordjylland, Nordfyn og Nordsjælland, vi sluttelig lagde os fast på. At valget faldt på disse tre regioner skyldtes, at de indbyrdes fremstod så dialektalt forskellige, at de virkede velvalgte for at kunne besvare vores problemformulering. Da den ovenstående selektering var foretaget, nåede vi frem til et undersøgelsesmateriale med en sammensætning som vist i nedenstående skema.

	Nordfyn	Nordjylland	Nordsjælland
	Mænd	Mænd	Mænd
16 – 30 år	21	39	64
31 – 45 år	32	47	65
46 – 60 år	15	39	68
I alt	68	125	197

2.3. F´-analysen

Et uundværligt redskab til at gennemføre vores undersøgelser har været analysemetoden F'. Med F'-analysen har det været muligt at udregne formantværdierne og efterfølgende indsætte dem i et koordinatsystem. Vi har valgt at indsætte F1-værdien på den ene akse og et gennemsnit af F2 og F3-værdierne på den anden akse og på den måde få mulighed for at sammenligne to taleres udtale af et fonem. Til analysen er programmellet Waves+ Product release 5,3, Entropic Inc. blevet anvendt.

BEST COPY AVAILABLE

2.4. Kontrol af analysematerialet

Inden de udvalgte fonemer blev omsat til F´-plot, blev de kontrolleret auditivt såvel som visuelt. Kontrollen foregik ved, at vi gennemlyttede de sætninger, der senere skulle analyseres for at sikre, at lydsiden stemte overens med transskriptionen. Samtidig kunne vi frasortere sætninger med diegetisk støj eller andre elementer, der kunne virke forstyrrende på analysen.

Vi kontrollerede endvidere visuelt, at der ikke var sammenfald i formantkurverne, som blev estimeret for hvert enkelt fonem. Forekom der sammenfald, betød det, at programmellet havde fejlestimeret formantværdierne, og den pågældende analyse blev kasseret.

3 Resultaterne af den teknisk analyse

3.1. F´-plottene

De fremanalyserede F´-plot blev placeret i tre separate koordinatsystemer for at få et samlet billede af et eventuelt mønster i hver aldersgruppe. Herunder ses således det koordinatsystem, der viser F´-plottene for gruppe et, altså de 16 til 30-årige. De bogstaver der markerer plottene refererer til de udvalgte regioner. Bogstavet j betyder Nordjylland, f Nordfyn og z Nordsjælland. Som det fremgår af nedenstående koordinatsystem ligger plottene temmelig spredt. Plottene for F1 ligger således fra 350-950 Hz mens de for F2´ ligger fra 950-2250 Hz. Vi har dog afgrænset et kerneområde, der for F1 cirka ligger mellem 525 og 700 Hz og for F2´ mellem 1700 og 2000 Hz. Af pladshensyn har vi ikke her valgt at medtage de to resterende koordinatsystemer for gruppe to og tre, men vil naturligvis angive spredning såvel som kerneområder.

For gruppe to, der bestod af 31 til 45-årige, fordelte plottene sig fra 250 til 900 Hz på F1-aksen og fra 1100 til 2200 Hz på F2´-aksen. Kerneområdet for gruppe

to blev fra cirka 500 til 700 Hz på F1-aksen og fra cirka 1700 til 2000 Hz på F2´-aksen.

I gruppe tre, der udgjordes af de 46 til 60årige, var plottene fordelt mellem 275 og 900 Hz på F1-aksen og 950 og 2300 Hz på F2´-aksen. Kerneområdet var mellem cirka 600 og 800 Hz på F1-aksen og 1700 og 1900 Hz på F2´-aksen.

Når man sammenligner de tre gruppers kerneområderne, ses et vist sammenfald, hvilket dog tydeligst gør sig gældende for gruppe et og to. Ved at behandle de værdier, F´-plottene byggede på, efter en række statistiske metoder var det dog muligt at sammenligne værdierne på en mere tydelig og præcis måde, hvilket kan ses i det følgende.

3.2. Middelværdier, standardafvigelser og T-tests

Middelværdien er udregnet ved at addere værdierne for samtlige forekomster på hver akse og dividere dem med antallet af forekomster. Herved har det været muligt at se tendenser i realisationerne i grupperne, men denne vurdering bør dog ikke fortages uden slutteligt at sammenholde middelværdierne med standardafvigelserne.

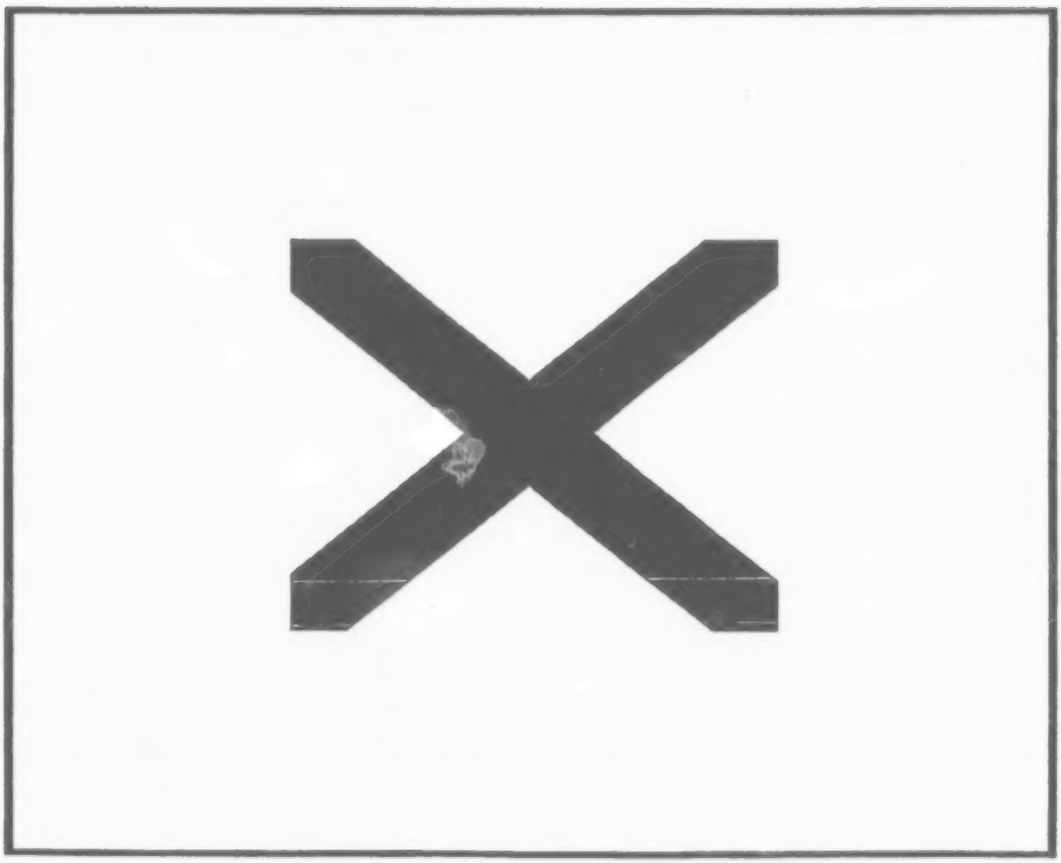
Standardafvigelsen er et udtryk for hvor stor spredningen er mellem de værdier, som middelværdien er udregnet på baggrund af. Dermed får man et udtryk for pålideligheden af det billede, middelværdien giver af realisationerne.

Den sidste metode, vi har benyttet os af for at vurdere resultaterne af F´-analysen, er en T-test; en beregning af sandsynligheden for sammenfald af formantværdierne i to grupper, hvorfor man også kan vurdere sandsynligheden for sammenfald i realisationen af [A:] i de forskellige regioner.

På de næste sider ses to tredimensionale søjlediagrammer, der viser middelværdierne for henholdsvis F1 og F2´ på tværs af regionerne og aldersgrupperne. Skemaerne, der findes under søjlediagrammerne, indeholder standardafvigelserne for F1 og F2´ i de tre regioner og aldersgrupper.

Standardafvigelse

F1	NJ	NZ	NF
G1	122,6	109,3	117,6
G2	121,1	129,8	90,5
G3	127,7	121,6	92,2



Standardafvigelser

F2'	NJ	NZ	NF
G1	119,2	150,4	148,3
G2	134,5	140,5	96,7
G3	181,8	129,3	110,1

T-test

		F1	F2'
G1	Z – F	29%	77%
	J – F	94%	18%
	J –Z	17%	20%
G2	Z – F	67%	34%
	J – F	21%	71%
	J – Z	8%	56%
G3	Z –F	35%	81%
	J – F	82%	73%
	J – Z	38%	78%

Ved at betragte diagrammerne med F'-plottene for alle tre aldersgrupper var det muligt at bestemme kerneområderne omtrentligt. Søjlediagrammerne over middelværdierne viser separat visse tendenser, selv om det er svært at påpege nogen tydelig relation mellem F1- og F2'-middelværdierne. Inden for F1-middelværdierne var der tre tendenser. Den tydeligste tendens var, at de nordsjællandske middelværdier var lavest i samtlige aldersgrupper. Endvidere havde Nordjylland generelt høje middelværdier. Denne tendens blev dog forstyrret af Nordfyns gruppe 3, der havde den højeste middelværdi i F1. Desuden lå middelværdierne for gruppe 3 generelt højere end de øvrige. Et billede, der kun blev

ødelagt af Nordjyllands gruppe 2, hvis middelværdi lå få Hz over samme regions gruppe 3.

Standardafvigelserne for F1-værdierne er generelt høje, især de nordjyske samt Nordsjællands gruppe to og tre. Dette betyder ikke nødvendigvis, at de nævnte gruppers F1 generelt er lave på trods af den lave middelværdi. Omvendt er standardafvigelserne for Nordfyn gruppe 2 og 3 forholdsvis lave, hvilket betyder, at middelværdierne giver et relativt pålideligt billede af realisationerne.

For F2' kan der ligeledes påvises visse tendenser inden for materialet. Middelværdierne for gruppe 1 lå markant højere end de øvrige F2'-middelværdier, og der var endvidere tendens til, at middelværdierne for gruppe 2 og 3 ikke afveg ret meget fra hinanden. Standardafvigelserne for F2' var generelt højere end standardafvigelserne for F1. På trods af dette forhold bør man anse tendenserne i F2' for at være mere pålidelige end tendenserne for F1, da standardafvigelsen også bør ses i forhold til den numeriske værdi den refererer til.

Ved at sammenligne standardafvigelserne for F1 og F2' var det muligt at påpege en klar tendens i materialet. Nordfyns gruppe 2 og 3 udmærkede sig i denne sammenhæng ved at have de laveste standardafvigelser overhovedet i både F1 og F2'. Derfor må disse to grupper betragtes som de relativt mest ensartede med hensyn til realisationen af [A:].

I T-testene fik vi et billede af sandsynligheden for sammenfald i realisationerne. Det er dog svært at se et gennemgående mønster i sandsynligheden for sammenfald mellem regionerne, når man sammenholder alle tre aldersgrupper. Der kan alligevel spores enkelte tendenser i resultaterne. Imellem Nordjylland og Nordsjælland er der således generelt en lav sandsynlighed for sammenfald i alle tre grupper. Det skal dog bemærkes at sandsynligheden for sammenfald er relativt høj i F2' mellem Nordsjælland og Nordjylland i gruppe 3. Dette ligger i forlængelse af en anden tendens i T-testene, nemlig at sandsynligheden for sammenfald er jævnt stigende fra gruppe 1 til gruppe 3.

Man må på baggrund af den tekniske analyse konkludere, at de forskellige behandlinger af data og sammenligningerne, der er blevet foretaget på baggrund af disse, har gjort det muligt at påvise en række tendenser i materialet. Disse tendenser skal dog tages med forbehold, ligesom man må konstatere, at det ikke var muligt at se tydelige endsige entydige regionale eller aldersbestemte mønstre i materialet.

4. Mulige fejlkilder

Vi har i analysen konstateret visse tendenser, men disse, såvel som analysematerialet som sådan, bør dog vurderes ud fra visse forhold, der har kunnet tænkes at påvirke deres udseende. Som tidligere nævnt har vi i udgangspunktet forsøgt at eliminere eller mindske de faktorer, som materialet har kunnet være under indflydelse af.

Når resultatet alligevel blev så disparat som det er fremgået af analysen, bør man overveje, hvilke andre faktorer der har kunnet tænkes at have indvirket på resultatet.

Det ville umiddelbart være nærliggende at antage, at en del af spredningen kan tillægges fejl i analyseprogrammellet. Gennem prøveanalyser af materialet har det dog vist sig at alvorlige fejl i søgealgoritmen tydeligt ses i F'-diagrammet. Endvidere var det muligt at konstatere et vist fællestræk i grupperingen af plot for alle tre grupper samt at bestemme kerneområder for disse der desuden ligger i relativ nærhed af det estimat for [A:], som blandt andre Grønnum og Thorsen angiver.

En faktor i materialet, som ligeledes kan have været udslagsgivende for formantværdierne, er stød. Der er i transskriptionen af de i SpeechDat 2 optagne sætninger ikke taget højde for stød. Denne mangel blev først klar, efter analysen var foretaget, hvorfor det ikke var muligt efterfølgende at kompensere for dette.

Det har derfor heller ikke været muligt at vurdere, hvor stor indvirkning dette forhold har haft på analyseresultaterne.

Derudover kan resultatet være blevet forstyrret af det forhold, at de medvirkende talere selv har skullet indplacere sig i forhold til deres egen vurdering af deres udtales regionale tilhørsforhold. Man kunne dermed forestille sig, at en del af talerne hverken har haft det fornødne kendskab til sine egne eller regionernes dialektale karakteristika.

Et andet forhold ved indsamling af data til SpeechDat 2, som må forventes at have præget udtalen, er, at hovedparten af de ord og sætninger som personerne har indtalt, var fortrykte på det ark, de fik tilsendt. Visse talere har derfor utvivlsomt forsøgt at tillempe en udtale, som tydeligvis ikke har været dem naturlig, men svarede til en opfattelse af rigsdansk.

En anden årsag til, at regionernes [A:]-realisationer fremtræder så uensartede, kunne være, at der i SpeechDat 2 ikke er taget forbehold for sociolektale variationer. Man kunne med andre ord forestille sig, at der i materialet findes realisationsmønstre for sociale grupper inden for eller på tværs af områderne, som kunne få regionen til at fremstå endnu mere uhomogen. Desuden må man formode, der inden for de respektive regioner kan findes dialektale realisationsvariationer, som ligeledes vil forstyrre det samlede billede for udtale. Dette kunne pege i retning af, at regionerne måske ikke i alle henseender var hensigtsmæssigt afgrænsede.

5. Konklusion

Vi har ved hjælp af den foregående analyse været i stand til at påvise visse tendenser i materialet. Når man betragter de udledte tendenser samlet, kan de i visse tilfælde underbygges af hinanden og i andre tilfælde forringes. I F1-middelværdierne påpegede vi tendenser til, at de nordsjællandske værdier lå lavere og de nordjyske generelt højere. Dette underbygges til dels af T-testene, der

viser, at der netop mellem Nordjylland og Nordsjælland er en lav sandsynlighed for sammenfald i realisationerne. Man kunne derved umiddelbart forledes til at antage, at der mellem disse to regioner var et tydeligt mønster i realisationerne inden for disse. Ved at sammenholde resultatet med standardafvigelserne viser det sig, at den regionale forskel ikke kan siges at bygge på en klart regional realisationsforskel.

Indenfor F2'-middelværdierne påviste vi en aldersmæssig tendens i materialet. Gruppe 1 lå højest og gruppe 2 og 3 tæt på hinanden. Dette kunne umiddelbart tyde på, at man kunne efterspore aldersbestemte fællestræk i realisationen af [A:] for F2'. Denne antagelse kunne til en vis grad underbygges af T-testene, idet vi i disse påviste en tendens til at sandsynligheden for sammenfald var stigende i forhold til gruppens aldersniveau. Man kunne gå så langt som til at fremsætte en hypotese om, at når gruppe 1 har de højeste F2'-middelværdier, kunne det være et tegn på en aldersbestemt udvikling og ændring af realisationen af [A:] i F2', hvilket igen kunne underbygges af at de øvrige to gruppers middelværdier lå så tæt. Tendensen til en sådan udvikling må dog alene på baggrund af standardafvigelserne betegnes som, om ikke fejlagtig, så i hvert fald yderst vag.

Den eneste tendens i analysen der peger i retning af en vis grad af regional ensartethed i realisationen af [A:] er at Nordfyns gruppe 2 og 3 har de laveste standardafvigelser i både F1 og F2'. Denne tendens må dog igen sammenholdes med det faktum, at standardafvigelserne generelt er høje, hvorfor det i det hele taget er vanskeligt at tale om regional homogenitet i realisationen, når man ser på standardafvigelserne.

Ved at gennemføre analysen har vi dog kunnet besvare vores problemformulering, da det har været muligt at påvise regionale og aldersbestemte tendenser i materialet.

Resultaterne er dog som nævnt præget af en vis usikkerhed, hvorfor det ikke er muligt at konkludere, at der tegner sig et overbevisende billede af regionale eller aldersmæssige mønstre i realisationen af fonemet [A:]. Det ville dog være rimeligt at antage, at det havde været muligt at få et andet og måske tydeligere analyseresultat, hvis der havde været taget højde for de eventuelle fejlkilder og mangler, der er blevet anført ovenfor.

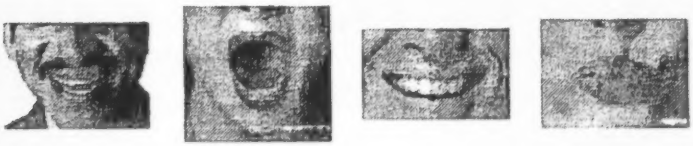
6. Litteratur

Brink, Lars, et al. 1974. *Udtaleforskelle i Danmark.* København:Gjellerup

Grønnum, Nina, Oluf Thorsen. 1991. *Fonetik for sprogstuderende.* København:Københavns Universitet, Institut for Almen og Anvendt Sprogvidenskab.

Lindberg, Børge. 1998. SpeechDat Danish FDB 4000 Speaker Database for Fixed Telephone Network. *EU-Project SpeechDat, LE2-4001,* Version 1.0.

WHEN TALK IS A SCIENCE...



Linguistics and Language Behavior Abstracts

Comprehensive, cost-effective, timely coverage of current ideas in linguistics and language research

Abstracts of articles, books, and conference papers from nearly 1,500 journals published in 35 countries; citations of relevant dissertations as well as books and other media.

Available in print or electronically through the Internet Database Service from Cambridge Scientific Abstracts (*www.csa.com*).

Contact *sales@csa.com* for trial Internet access or a sample issue.

Linguistics & Language Behavior Abstracts

Published by Cambridge Scientific Abstracts
7200 Wisconsin Avenue • Bethesda MD 20814 • USA
Tel: +1 301-961-6700 • Fax: +1 301-961-6720 • E-mail: *sales@csa.com*

BEST COPY AVAILABLE

Odense Working Papers
in Language and Communication

OWPLC publishes scholarly articles written by members of the Institute of Language and Communication at the University of Southern Denmark and their Danish or foreign colleagues.

Nos 17-24 include:

No. 17 Dorthe Bleses:
The role of input, productivity and transparency in Danish children's acquisition of past tense morphology

No. 18 Teresa Cadierno (ed.):
L2 Listening comprehension

No. 19 Carl-Erik Lindberg & Steffen Nordahl Lund (eds.):
17th Scandinavian Conference of Linguistics, vol I-II
20-22 August 1998

No. 20 Thomas O. Madsen:
Sproglyde: fysiske kontinua og mentale kategorier

No. 21 Thomas Lund Andersen:
Stjernernes tale – register i teori og praksis

No. 22 Johannes Wagner (udg.):
Pædagogik og læring I fremmed- og andetsprog

No 23 Anders Hougaard & Steffen Nordahl Lund (eds.):
The Way We Think, vol I-III

No 24 Poul Søren Kjærsgaard (ed.):
Datalingvistik 2000

Copies can be obtained from

Institute of Language and Communication
University of Southern Denmark,
Campusvej 55
DK-5230 Odense M.
Denmark

General Editor: Steffen Nordahl Lund
(E-mail: snl@language.sdu.dk.)

Odense Working Papers in Language and Communication publishes scholarly articles written by members of the Institute of Language and Communication at the University of Southern Denmark, and their Danish or foreign colleagues.

This issue of ***Odense Working Papers in Language and Communication*** consists of six papers presented at the 10th Annual Meeting of Datalingvistisk Forening, held in Odense in May 2000.

The main topic of the meeting was some different applications of a number of different applications of speech processing.

The paper presented by Laila Dybkjær and Niels Ole Bernsen describes the MATE project and its work towards defining standards for speech corpus annotations.

Charles Hoequist presents an account of a Danish national project of Text-to-Speech Synthesis. The system relies on a database of encoded speech segments.

Gregers Koch accounts for a method of designing automatically semantic representations of information, especially certain computational linguistic problems.

Thomas Madsen, Hans Basboll and Claus Lambertsen present current work on the development of a semi-automatic morphological and phonetic program intended for coding written Danish texts.

Bolette Sandford and Sanni Nimb present a model for coding information pertaining to Danish verbs, especially those which combine a verbal form with a preposition or an adverbial particle.

Jacob Teckemeier and Kim Hornbech analyse different realisations of the [A:] phoneme in Danish, relative to variation in age and geography.

UNIVERSITY OF SOUTHERN DENMARK

BEST COPY AVAILABLE

100

U.S. Department of Education
Office of Educational Research and Improvement (OERI)
National Library of Education (NLE)
Educational Resources Information Center (ERIC)

NOTICE

Reproduction Basis

[X] This document is covered by a signed "Reproduction Release (Blanket)" form (on file within the ERIC system), encompassing all or classes of documents from its source organization and, therefore, does not require a "Specific Document" Release form.

[ ] This document is Federally-funded, or carries its own permission to reproduce, or is otherwise in the public domain and, therefore, may be reproduced by ERIC without a signed Reproduction Release form (either "Specific Document" or "Blanket").